DOCUMENT RESUME

ED 052 493 24 CG 006 503

AUTHOR Roman, Richard Allan
TITLE The Script Network: Present Conditions and Suggestions. Information System for Vocational Decisions.
INSTITUTION Harvard Univ., Cambridge, Mass. Graduate School of Education.
SPONS AGENCY Office of Education (DHEW), Washington, D.C.
REPORT NO Proj-R-31
BUREAU NO BR-6-1819
PUB DATE Mar 70
GRANT OEG-1-6-061819-2240
NOTE 130p.

EDRS PRICE EDRS Price MF-$0.65 HC-$6.58
DESCRIPTORS Career Choice, *Computer Assisted Instruction, Computers, Educational Technology, *Information Networks, Information Systems, Instructional Media, Language, *Man Machine Systems, *Occupational Guidance, Programing, Vocational Development
IDENTIFIERS Information System for Vocational Decisions

ABSTRACT

The overall organization of the Information System for Vocational Decisions (ISVD) script network is diagrammed and each component described: (1) systems scripts which help determine what part of the system the inquirer wants to use; (2) CAI scripts designed to teach the inquirer things related to career choice; (3) preference scripts which help transform value preferences about criteria for work or school into concrete alternatives; (4) data scripts which allow direct exploration of the data bases; (5) templates which are similar to, but less specific than, the data retrieval scripts; and (6) access routines which represent various components of the ISVD paradigm of vocational decision-making and are designed to help the inquirer become conscious of himself as a decider. This paper reviews each of these, except the CAI scripts, and suggests improvements. (TL)

ED052493

INFORMATION SYSTEM FOR VOCATIONAL DECISIONS

Project Report No. 31

THE SCRIPT NETWORK:

PRESENT CONDITIONS AND SUGGESTIONS

Richard Allan Roman

This paper was supported in part by Grant No. OEG-1-6-061819-2240
of the United States Office of Education under terms of the
Vocational Education Act of 1963.

Graduate School of Education
Harvard University

March 1970

U.S. DEPARTMENT OF HEALTH, EDUCATION & WELFARE
OFFICE OF EDUCATION
THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL OFFICE OF EDUCATION POSITION OR POLICY.

CG 006 503

TABLE OF CONTENTS

Overview of the ISVD Network 1
The First Interaction 8
  Introduction to System 10
  Introduction to Commands 11
ORIENTI the major orientation 12
  A Typical Overview Script 15
Minor Orientations 19
  The Data Script -- access to storage 20
  QUIT -- end for the day 50
  STOP -- end of any activity 51
  SUMMARY -- review of an interaction 51
New Student Commands 53
#(SCR)FACT Vectors -- coding of factors affecting decisions 55
ANALYZE -- encoding English into #(SCR)FACT vectors 62
Reality Feedback 64
  Template -- description of data items 64
EXPLORONE -- general analysis of a discontinuity 69
GOOD OR BAD ITEM -- valuing alternatives 79
COLCHANGEFACTOR -- monitoring apparent value changes 84
SELF CONCEPTS PROFILING TECHNIQUE -- a procedure for accessing
        concepts of self 89
LIFE CAREER GAME 95
EDIT-SEDIT -- script to help edit other scripts 101
SHOMP-HOMP -- setting home priorities 108
JOB CORPS READING TEST 108
COST -- highly interactive script on college costs 110
Preference Scripts 119

The SCRIPT NETWORK:

Present Conditions and Suggestions

The overall organization of the script network is diagrammed in Chart 1, Script Network of ISVD. Each component is described in Table 1. Chart 1 is a compression of the actual network. Each of the orientations represents all the components attached to college. The CAI scripts box represents a large set of scripts. Chart 2 more accurately represents the occupational CAI network and Chart 3 represents the college CAI network. Chart 4 is the less elaborated trade school CAI network and Chart 5 is the military CAI network.

Script Network of ISVD

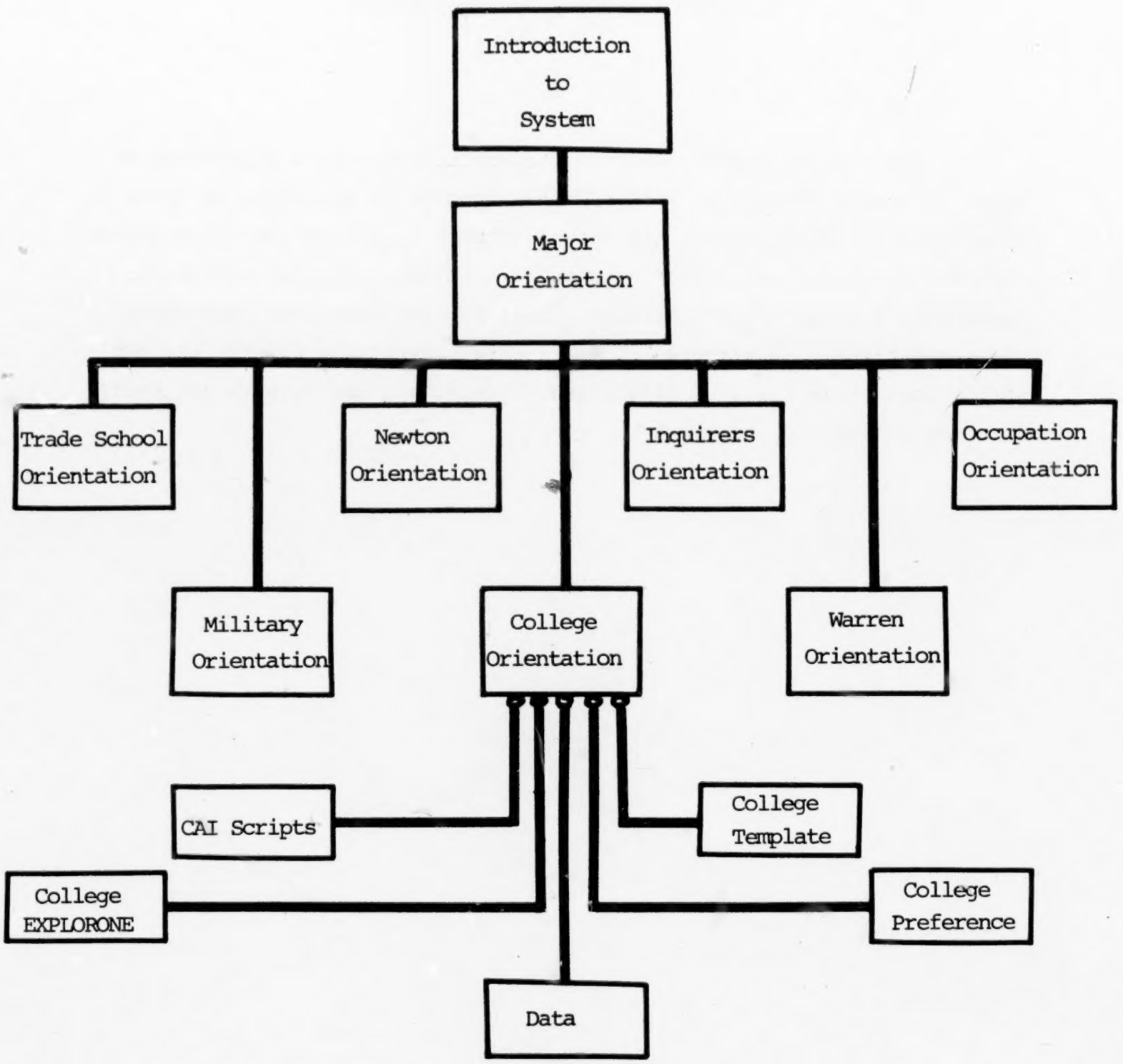
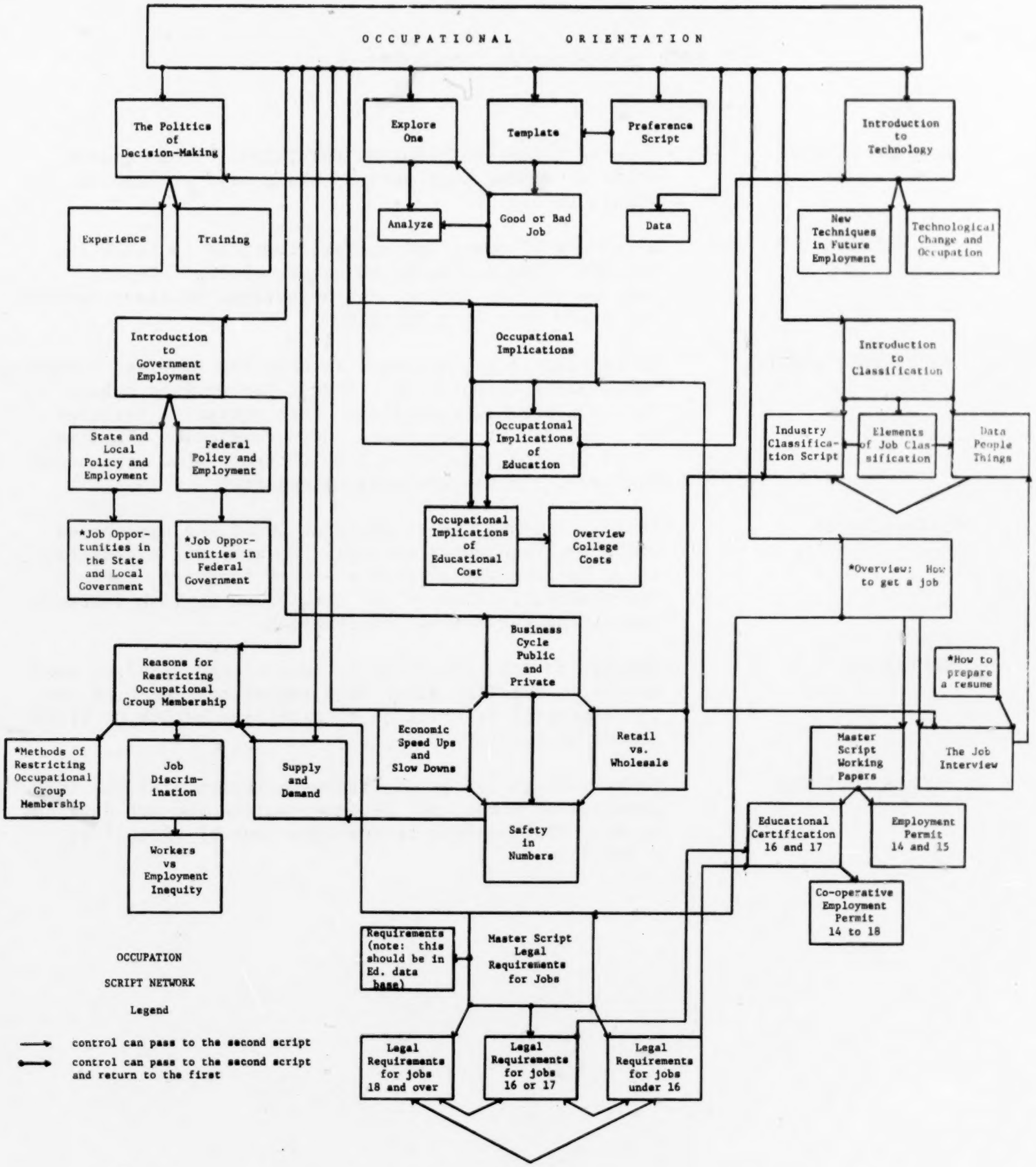


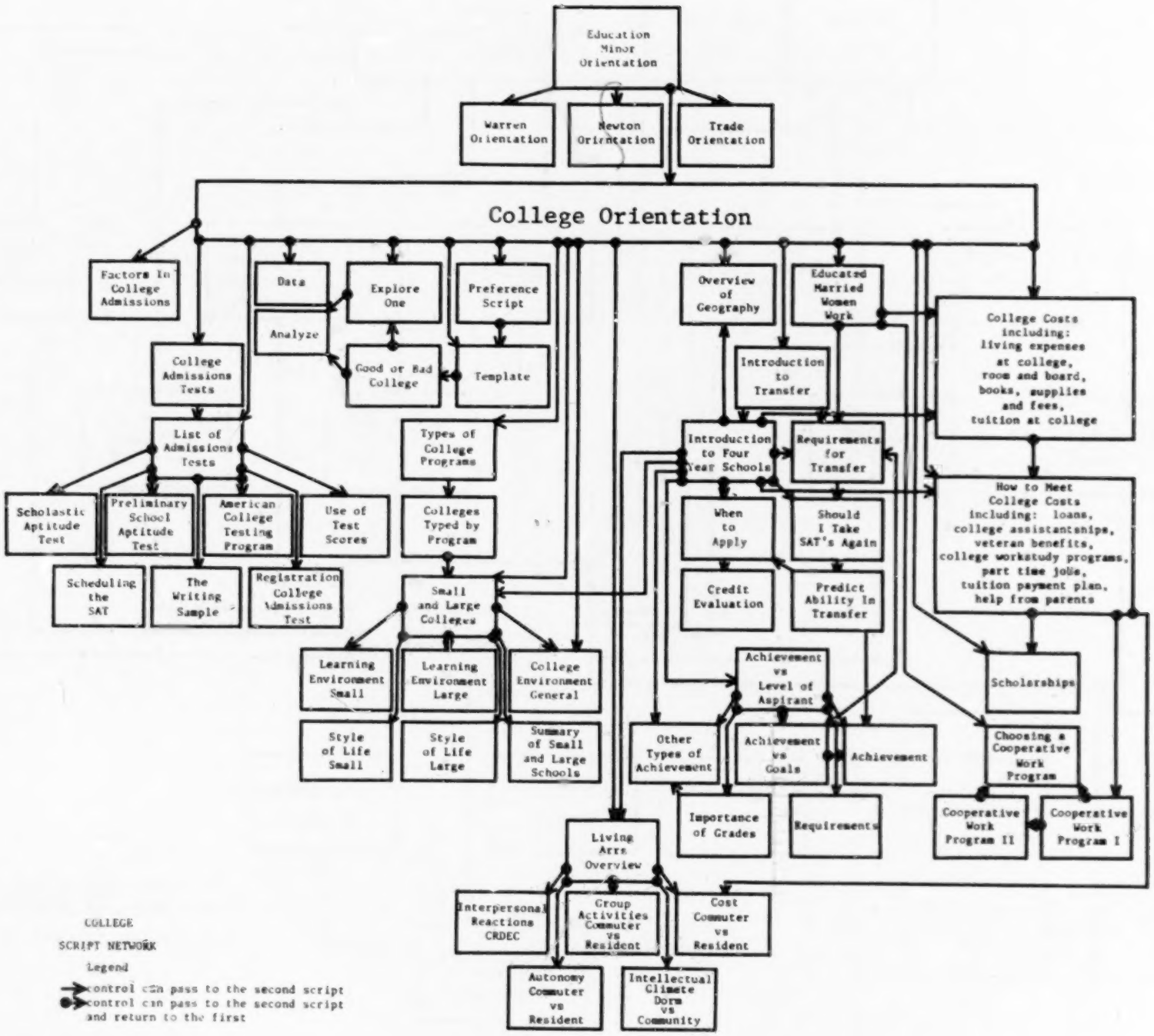
Chart 1

ISVD SYSTEM SCRIPT COMPONENTS

SYSTEMS SCRIPTS	Such as MAJOR and MINOR ORIENTATIONS. This network helps determine what part of the system an inquirer wishes to use.
CAI SCRIPTS	A network of about 16Ø scripts designed to teach the inquirer things related to career choice. Topics include college choice, job selection, military careers, and trade schools selection.
PREFERENCE SCRIPTS	These scripts are designed to help the inquirer transform value preferences about criteria for work or school into concrete alternatives. Each script is tailored to a particular data base. Upon completion there is filed for the inquirer a list of those jobs or schools that meet all the criteria he selected.
DATA SCRIPTS	This set allows direct exploration of the data bases under the inquirer's control. He asks for a specific datum by name and attribute -- for instance, Harvard University, Tuition -- and obtains an English sentence containing the desired information.
TEMPLATES	Similar to data retrieval scripts except inquirer need not be as specific about data requested. Here he can get a general description of a college or job or trade school or military specialty.
ACCESS ROUTINES	These scripts represent various components of the ISVD paradigm of vocational decision-making and are designed to help the inquirer become conscious of himself as a decider.

Table 1


OCCUPATIONAL ORIENTATION
The Politics of Decision-Making
Explore One
Template
Preference Script
Introduction to Technology
Experience
Training
Analyze
Good or Bad Job
Data
New Techniques in Future Employment
Technological Change and Occupation
Introduction to Government Employment
Occupational Implications
Introduction to Classification
State and Local Policy and Employment
Federal Policy and Employment
Occupational Implications of Education
Industry Classification Script
Elements of Job Classification
Data People Things
*Job Opportunities in the State and Local Government
*Job Opportunities in Federal Government
Occupational Implications of Educational Cost
Overview College Costs
*Overview: How to get a job
Business Cycle Public and Private
*How to prepare a resume
Reasons for Restricting Occupational Group Membership
Economic Speed Ups and Slow Downs
Retail vs. Wholesale
Master Script Working Papers
The Job Interview
*Methods of Restricting Occupational Group Membership
Job Discrimination
Supply and Demand
Safety in Numbers
Educational Certification 16 and 17
Employment Permit 14 and 15
Workers vs Employment Inequity
Co-operative Employment Permit 14 to 18
Requirements (note: this should be in Ed. data base)
Master Script Legal Requirements for Jobs
Legal Requirements for jobs 18 and over
Legal Requirements for jobs 16 or 17
Legal Requirements for jobs under 16
OCCUPATION
SCRIPT NETWORK
Legend
control can pass to the second script
control can pass to the second script and return to the first



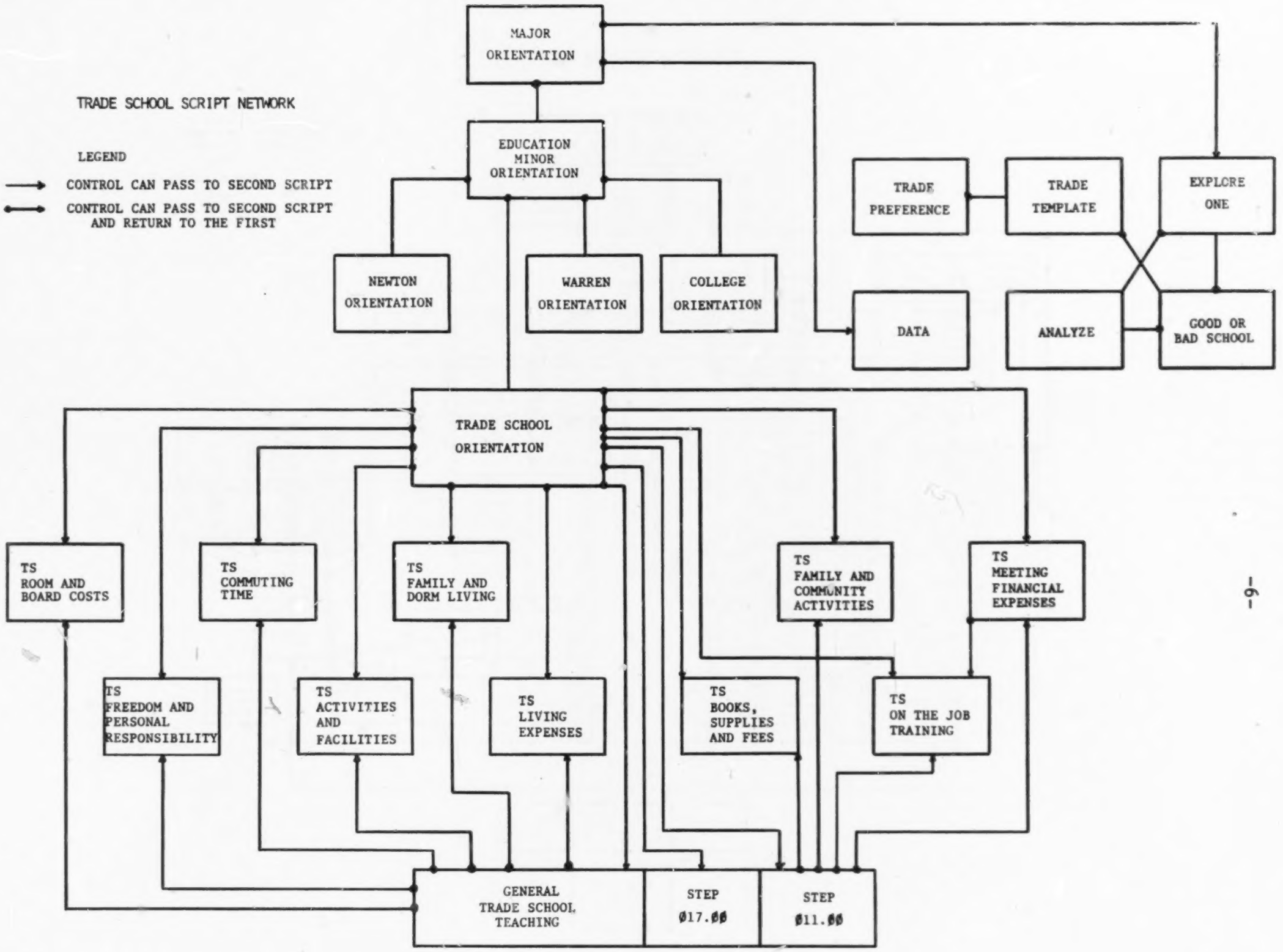
College Script Network

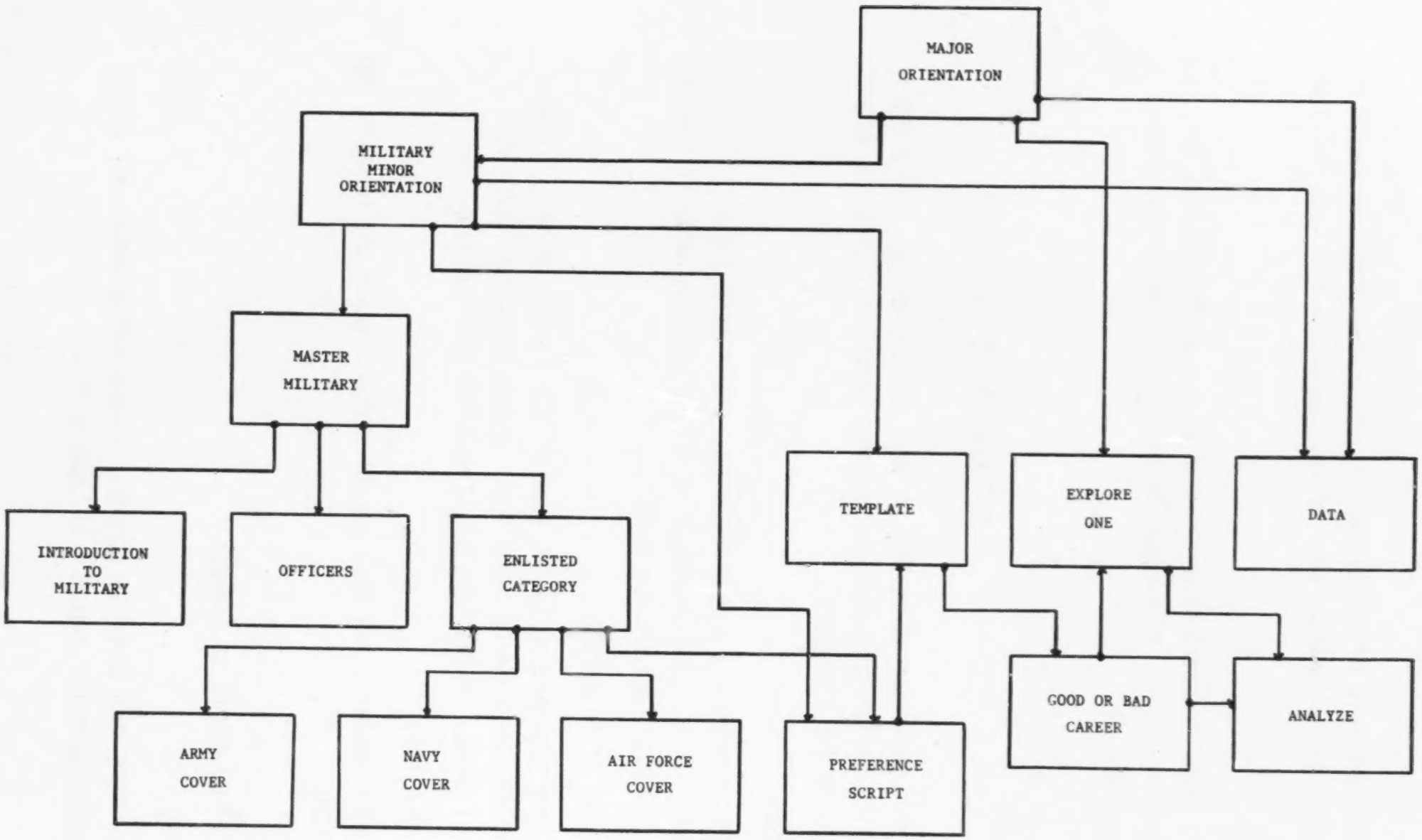
TRADE SCHOOL SCRIPT NETWORK

LEGEND

→ CONTROL CAN PASS TO SECOND SCRIPT

⟷ CONTROL CAN PASS TO SECOND SCRIPT AND RETURN TO THE FIRST





MILITARY SCRIPT NETWORK

LEGEND

→ CONTROL CAN PASS TO SECOND SCRIPT

⟷ CONTROL CAN PASS TO SECOND SCRIPT AND RETURN TO THE FIRST

This paper attempts to review each of the good scripts in the ISVD network and suggest improvements. None of the typical CAI scripts are discussed.

The ISVD script network always begins with the LOGIN script and several introductory scripts. A typical first interaction is represented by the following protocol.

INTRODUCTION TO THE SYSTEM

SYSTEM: PLEASE TYPE YOUR ID NUMBER SO THAT YOU CAN BEGIN. PRESS THE BUTTON MARKED SEND BLOCK WHEN YOU FINISH. (IT'S ON THE RIGHT SIDE OF THE KEYBOARD.)

INQUIRER: 0001

SYSTEM: YOUR ID NUMBER HAS BEEN RECEIVED. NOW PLEASE TYPE YOUR PASSWORD. IT WILL BE INVISIBLE SO NO ONE ELSE CAN READ IT. PRESS SEND BLOCK WHEN YOU ARE DONE.

INQUIRER: (TYPES IN PASSWORD)

SYSTEM: BEFORE BEGINNING WITH THE REST OF THE SYSTEM, WE NEED TO KNOW CERTAIN BASIC FACTS ABOUT YOU. PLEASE TYPE YOUR FIRST AND LAST NAMES. PRESS SEND BLOCK WHEN YOU ARE FINISHED.

INQUIRER: ROBERT DAVIS

SYSTEM: THANKS ROBERT. NOW YOUR AGE IN YEARS. PLEASE USE NUMBERS, NOT WORDS.

INQUIRER: 16

SYSTEM: ARE YOU A MALE OR FEMALE?

INQUIRER: MALE

SYSTEM: NOW TYPE YOUR GRADE IN SCHOOL. PLEASE USE NUMBERS, NOT WORDS. SOPHOMORE = 10, JUNIOR = 11, SENIOR = 12.

INQUIRER: 11

SYSTEM: WELCOME TO ISVD, ROBERT. WHEN YOU SEE THREE ASTERISKS *** AT THE BOTTOM OF THE SCREEN, THE MACHINE IS READY TO RECEIVE YOUR RESPONSE. IF YOU DON'T FEEL LIKE SAYING ANYTHING, SIMPLY PUSH THE "SEND BLOCK" BUTTON. TRY THAT NOW.

INQUIRER: (PUSHES SEND BLOCK BUTTON)

SYSTEM: SOMETIMES WE WILL ASK A DIRECT QUESTION, BECAUSE WE NEED TO KNOW SOMETHING FROM YOU BEFORE GOING ON. WHEN WE DO, PLEASE TYPE IN WHAT YOU WANT TO SAY AND PRESS END BLOCK. IS THIS CLEAR SO FAR?

INQUIRER: YES

SYSTEM: YOU ARE ALMOST READY TO USE THE SYSTEM. THERE ARE JUST A FEW OTHER THINGS YOU SHOULD KNOW ABOUT. IF YOU SHOULD WANT A PRINTED COPY OF WHAT IS ON THE SCREEN, WAIT UNTIL YOU SEE THE THREE ASTERISKS AND THEN PUSH THE BUTTON MARKED COPY PAGE. WAIT UNTIL THE MATERIAL HAS BEEN COPIED. THEN TYPE IN YOUR RESPONSE AND "SEND BLOCK". TRY IT NOW.

INQUIRER: (PUSHES COPY PAGE AND SEND BLOCK BUTTONS)

SYSTEM: DID YOU GET A PRINT-OUT OF WHAT WAS ON THE SCREEN?

INQUIRER: YES

SYSTEM: GOOD. SOMETIMES YOU WILL NOT KNOW HOW TO ANSWER THE SYSTEM WHEN IT ASKS YOU A QUESTION. IN THIS CASE, YOU SHOULD TRY TYPING @HELP. TRY IT ONCE.

INQUIRER: HELP

SYSTEM: NO, YOU DID NOT TYPE THAT CORRECTLY. TYPE THE SYMBOL @ AND THEN HELP. DO NOT PUT A SPACE IN BETWEEN.

INQUIRER: @HELP

SYSTEM: FINE. SOMETIMES THE SYSTEM WILL GIVE YOU A HINT WHEN YOU ASK FOR HELP, BUT NOT ALWAYS. YOU CAN ALWAYS TRY, THOUGH.

...

SYSTEM: WHAT DO YOU WANT TO DO TODAY?

INQUIRER:

Not all features of these routines are visible in the protocol. The functions and faults are detailed here.

The script network begins with the LOGIN script which does the following:

A. Asks for an identification code, checks for legitimacy of code, defined currently as existing as a record name in the NICKNAME file.

B. Asks for a secret password. Because of the Sanders equipment, the symbols > < make text between them invisible. We use this feature for security on passwords. If the inquirer's input matches the password, his session is begun. If not, he has 2 more chances before he is dismissed.

C. Defines strings

NICK

SEX

NAME

AGE

GRADE

by reference to the BASIC record of the appropriate file. #(NICK). If no values are found, inquirer is asked to supply the information which is stored for future reference. Script linkage is established by defining LINK = ORIENTI Ø5Ø.ØØ

STOP = ORIENTI Ø5Ø.ØØ

D. Control is transferred to INTRODUCTION TO SYSTEM.

INTRODUCTIONTOSYSTEM (ITS)

1. For a first time inquirer, ITS provides a full description of all console buttons. More experienced inquirers can skip this script or get a review.
2. Coding here catches likely errors and corrects them. The general experience is that the materials are lengthy, but valuable. Inquirers who will use the system only once or twice should be verbally instructed on console use, and skip this script.

3. At the end of this script, the command

   #(SR:#(NICK):TIMESΔTHROUGH:SYSTEM:#(AD:1:#(FO:#(NICK):
   TIMESΔTHROUGH:SYSTEM::Ø)))

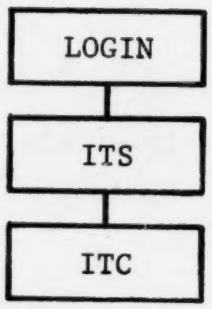
   records a successful login.

4. Control is passed to INTRODUCTION TO COMMANDS.

INTRODUCTIONTOCOMMANDS (ITC)

1. On the basis of number of times through the introduction to commands, a complete, summary, or optional introduction is supplied.
2. Control is transferred to ORIENTI.
3. ITC is confusing to most inquirers. They type a command rather than allow the machine to explain the command. This is undoubtedly a consequence of having just seen INTRODUCTIONTOSYSTEM where immediate response behavior is appropriate. A clear statement of expectation would help. Even better would be a consistent attitude between the two scripts.
4. Inquirers tend to think that only commands are appropriate responses, and use them too frequently at first.

In summary, the first scripts in any interaction are LOGIN, INTRODUCTIONTOSYSTEM, and INTRODUCTIONTOCOMMANDS. Logically they are arranged:



There are several problems with this arrangement from a network viewpoint.

1) A sophisticated inquirer can @STOP before the fact of his login is recorded. This causes problems in file access whenever he tries to fetch or store data in #(NICK).
2) Many inquirers learn to @STOP before ITC or ITS and therefore do not have their TIMES THROUGH record incremented for subsequent reference.

Once this behavior is established, the inquirer is always in the same logical session, and all sequence dependent data are confused.

3) The first use of LOGIN assumes knowledge taught in ITS and is therefore logically inconsistent.

RECOMMENDATIONS FOR CHANGES

A. Greater clarity of expectation is necessary for these introduction scripts. Most of the messages could be shorter and more to the point.

B. One possible tool, both here and elsewhere, is to have instructions of a general nature displayed in a separate block on the screen. Another possible approach is to use slides as instruction. Messages like "make a hardcopy," "Actually try that now," or "These instructions are just for information, you will use them later," would clarify expectations.

C. The logical organization would combine the three scripts as follows:
[see Chart 6 - Introduction to System on page 13]

D. Monitoring of behavior at the choice on introductory explanation could be used in several ways.

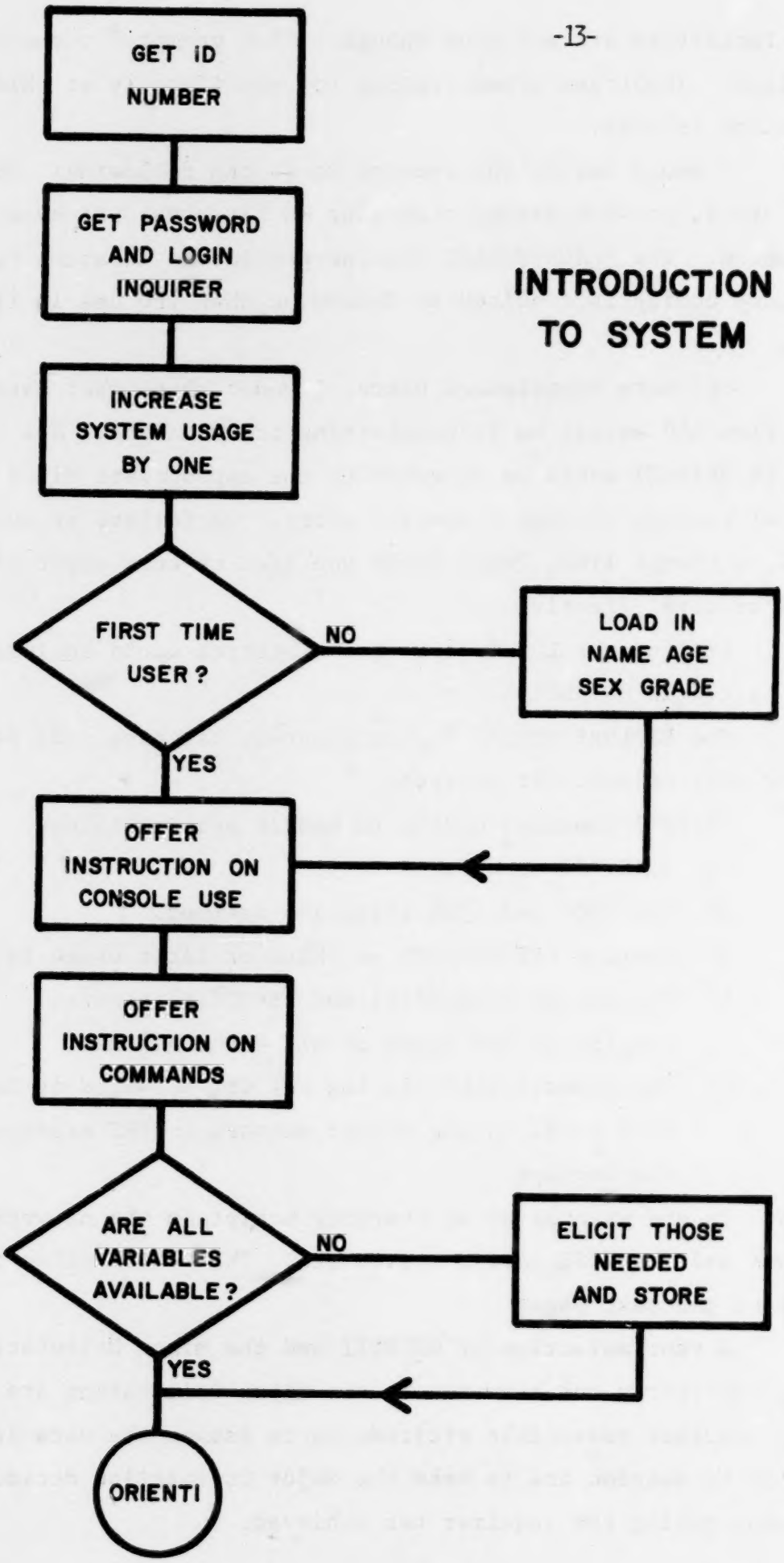
1. After two consecutive negative responses the question should be dropped. Possibly the major orientation should be modified to allow inquirer to ask for review if he wants it.
2. The inquirer's confidence, and personal investment in the system are probably related to his desire to review fundamentals. This could be used as a cue later, possibly to determine directiveness in other areas.
3. The cumulative responses across all inquirers could be totaled and used as feedback on efficiency of the routines. In ISVD we would use the INQTABLE file. You could go further here and differentiate groups of inquirers by age.

ORIENTI

This routine routes inquirers to the appropriate data base (minor orientation) or system routine.

Inquirer is expected to respond to the question, "What do you want to do?" Many inquirers have found this too general and non-directive. The

INTRODUCTION TO SYSTEM



HELP facilities are not good enough. Some prompted responses are not recognized. Inquirers often respond too specifically at this level and their intention is lost.

I would modify the routine to do the following: for first or second time users, provide strong prompting on first failure even if HELP is not requested. The MAJORSUGGEST routine provides a location for such prompting, but more coding is required to determine that the use is first or second time.

For more experienced users, I would check what data base he used last time and assume he is maintaining the same one. All responses that fail in ORIENTI would be directed to the appropriate Minor Orientation for further testing through a special entry. On failure at even the second level, a prompt like, "What would you like to know about colleges today?" could be more effective.

Note Listing the LG messages from logtapes would include all responses that failed in ORIENTI.

Note The EXPLORE DECOMP is too general, catching some detailed statements better sent to specific scripts.

ORIENTI contains coding to handle special things.

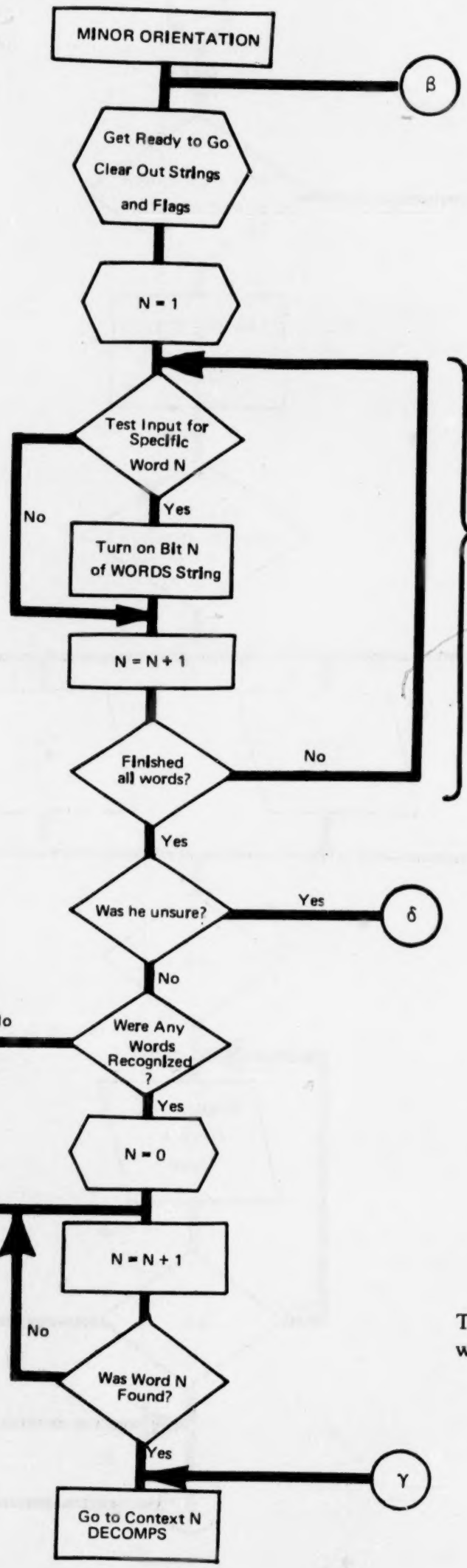
1) TLIST*** is defined.
2) The STOP and LINK lists are defined.
3) Testing for SUMMARY on third or later usage is performed.
4) Storage of STAKEN#(N) and DBVEC#(N) occurs.
5) Logging of the names of all forms defined.
6) The command #(LC) to log all CRT messages is issued. Up until this point in the script network no CRT messages are written to the logtape.

ORIENTI is one example of an overview script in the network. Minor Orientations and the COST script are others. These all follow the general flow chart on the next pages.

A reorganization of ORIENTI and the Minor Orientations seems desirable. Currently the keywords in the Major Orientation are according to data base. Another reasonable attitude is to assume the data is constant from session to session and to make the Major Orientation decide what stage of decision-making the inquirer has achieved.

A TYPICAL OVERVIEW SCRIPT

Assume at (β) that the keyboard buffer has the inquirer's last statement.



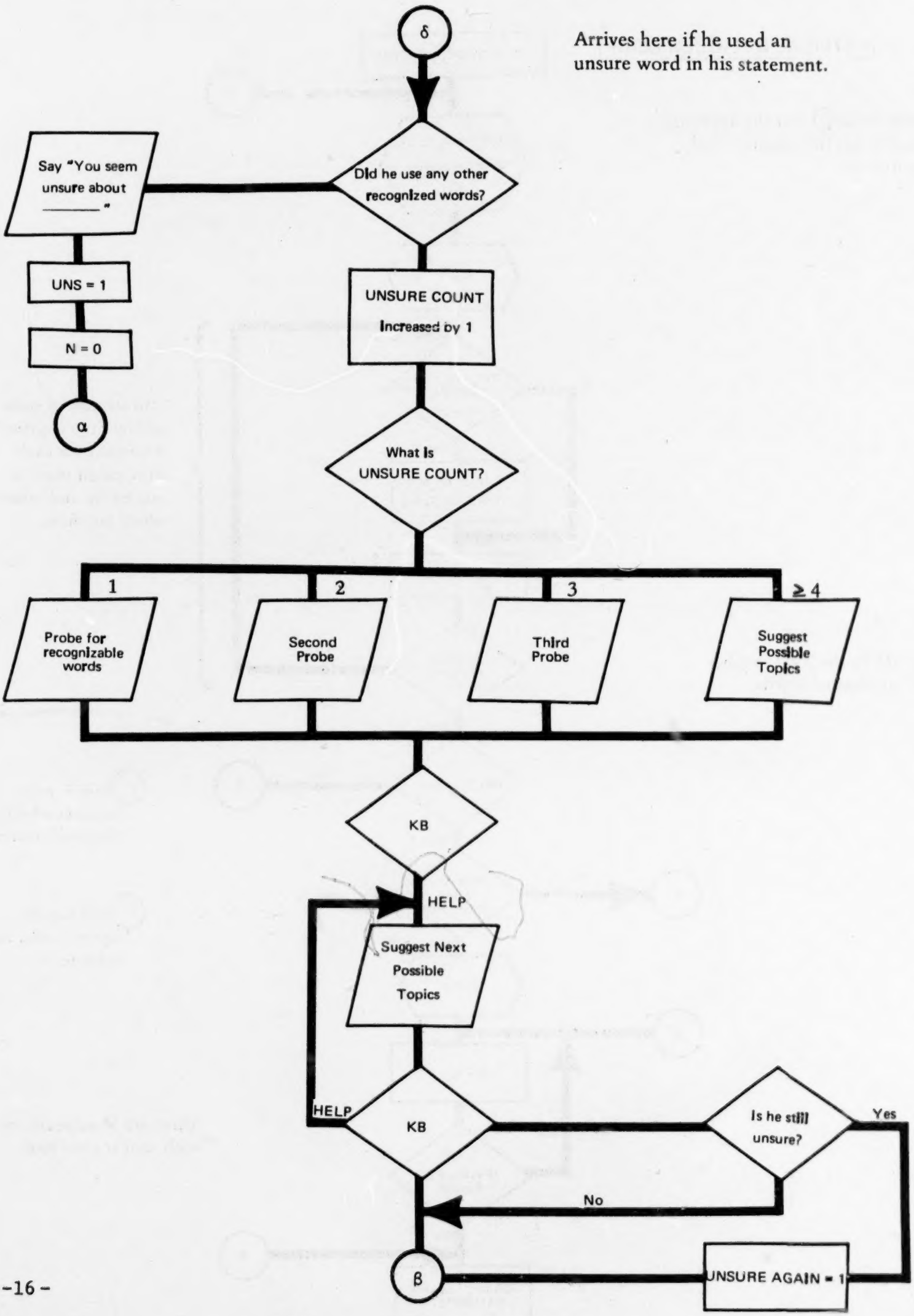
This section of code will test the inquirer's statement for each anticipated word in succession and remember which are there.

Let M be the total number of anticipated words.

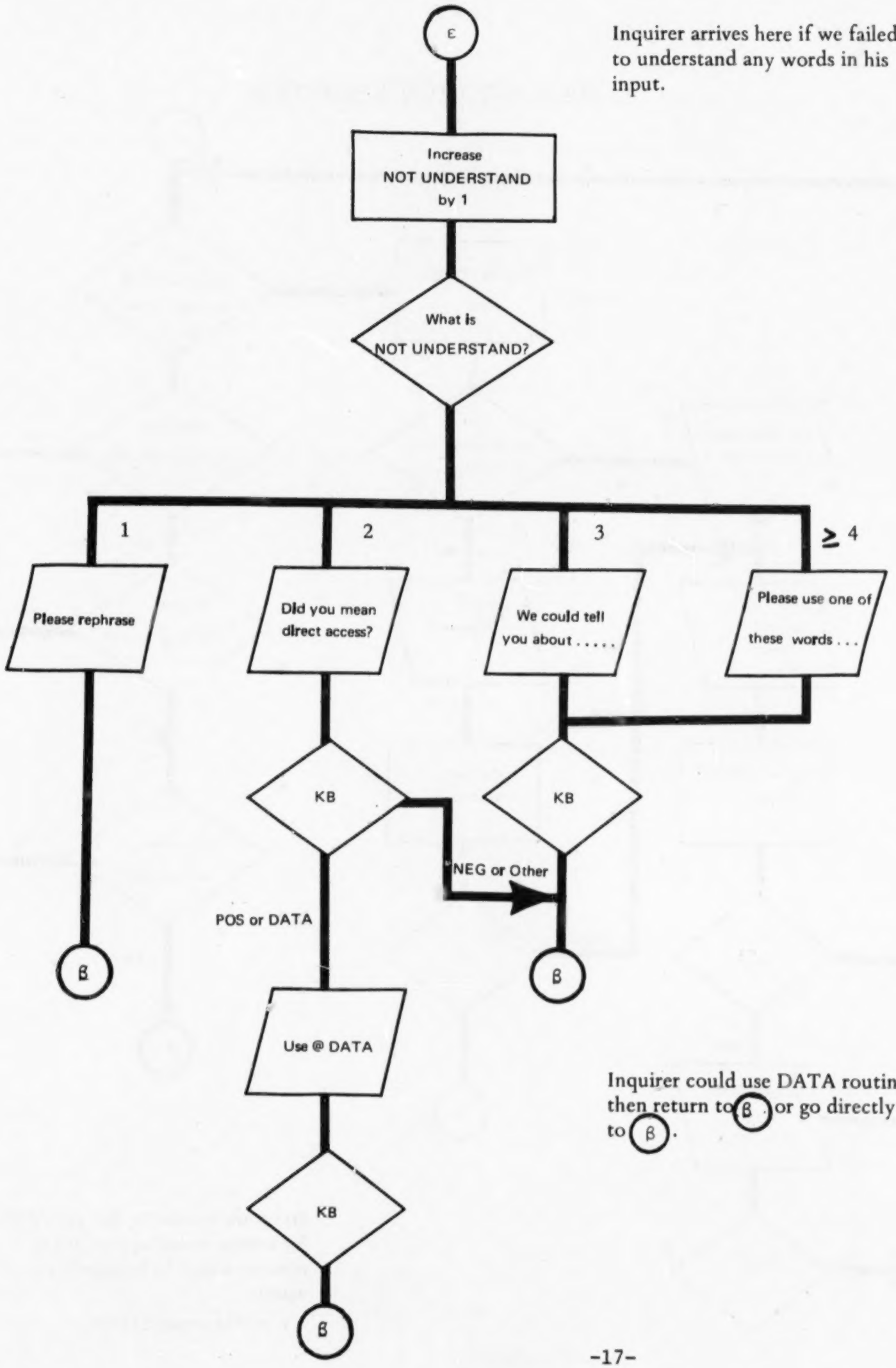
(δ) will handle inquirers who have expressed uncertainty.

(ε) will handle inquirers who weren't understood.

There are M subsections with similar structure.



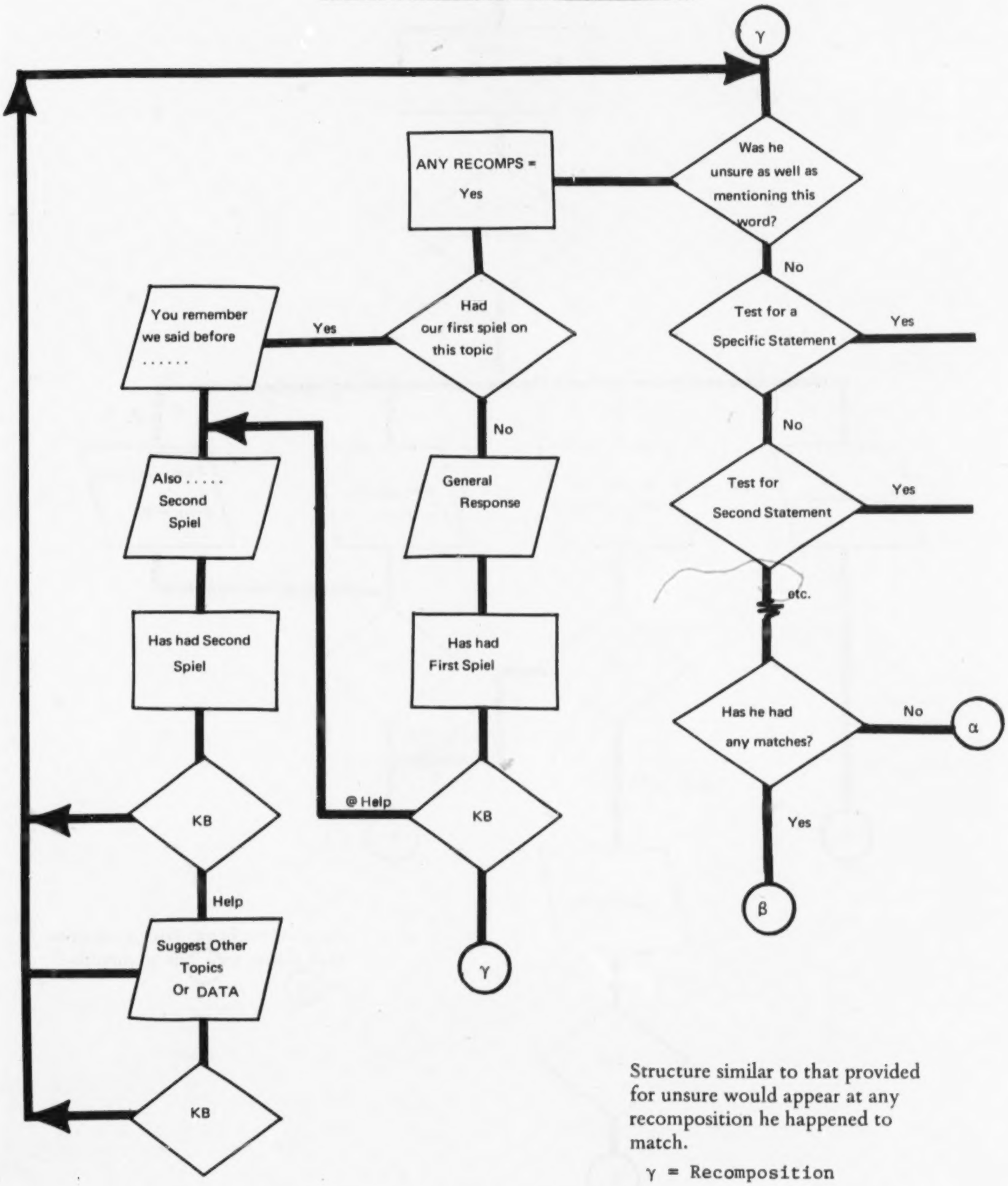
Arrives here if he used an unsure word in his statement.



Inquirer arrives here if we failed to understand any words in his input.

Inquirer could use DATA routines then return to (β) or go directly to (β).

A TYPICAL RECOMPOSITION SECTION



Structure similar to that provided for unsure would appear at any recomposition he happened to match.

γ = Recomposition

A typical interaction with ORIENTI would go as follows:

MACHINE: WHAT WOULD YOU LIKE TO DO?

INQUIRER: TELL ME ABOUT TUITION AT HARVARD.

MACHINE: WE HAVE NOT UNDERSTOOD YOUR REQUEST. IT WOULD HELP US IF YOU USE GENERAL, SIMPLE WORDS TO TELL US WHAT YOU WANT TO DO.

INQUIRER: HOW MUCH MUST I PAY TO GO TO HARVARD?

MACHINE: WE ARE STILL HAVING TROUBLE UNDERSTANDING YOUR REQUEST. SINCE WE WOULD LIKE TO MAKE THE MACHINE BETTER AT UNDERSTANDING PEOPLE, PLEASE PUSH THE HARD COPY BUTTON TO MAKE A PRINTOUT OF YOUR NEXT REQUEST.

INQUIRER: I AM SIMPLY TRYING TO ASK FOR INFORMATION ABOUT THE COST OF GOING TO HARVARD.

MACHINE: THE MACHINE'S LIMITED VOCABULARY INCLUDES THE WORDS -- DATA, MILITARY, SCHOOL, GAME, SELF, EDUCATION, JOB, SUMMARY, AND "DON'T KNOW". IF WHAT YOU WANT TO DO SEEMS RELATED TO ANY OF THESE WORDS, PLEASE TYPE THE WORD YOU THINK IS CLOSEST TO YOUR REQUEST.

INQUIRER: THANK HEAVENS. I WANT DATA ABOUT HARVARD.

MACHINE: YOU HAVE REQUESTED DATA ACCESS.

(inquirer is linked to the DATA routines.)

Minor Orientations

Minor orientations are specific to each data base. They are supposed to establish what an inquirer wants to do and send him to the appropriate script within the network for that data base. We have several levels of sophistication in our minor orientations. Some (the college and occupational) are designed to handle any English sentence and direct the inquirer to the appropriate script. The others require a multiple choice response or direct the course of interaction in a predetermined manner.

The multiple choice orientation is efficient getting the inquirer what he wants. It leaves no illusion of understanding, no mystery in the response. Our reason for having such orientations was the lack of time to prepare the natural language routines, however, multiple choice does have some advantages besides speed of preparation. Some of the most frustrating experiences of inquirers were in trying to get a minor orientation to direct them to the appropriate place.

The natural language orientations tried to use synonym lists to direct the inquirer. The inquirers were not told the appropriate level of response. Sometimes, when they requested help and got a list of possible typeins, the routine would not recognize even its own suggested words. This tended to make people angry.

The routines, when they did find a match in the statement usually behaved correctly. Most matches sent inquirers where they should have gone. However, the large number of failures to match indicate fundamental problems of conception or execution. Each failure to match is logged to the tape. Specifications for collecting these messages are included in the specifications for logtape processing.

DATA

The DATA script has a variety of functions which the following protocol illustrates.

DATA SCRIPT

INQUIRER: @DATA-COL

SYSTEM: THIS IS THE FIRST TIME YOU HAVE USED THE DATA ROUTINE ACCORDING TO THE RECORDS. IF YOU ARE REALLY AN OLD HAND AND WOULD LIKE TO SKIP THE INTRODUCTION, TYPE SKIP. OTHERWISE, JUST PRESS SEND BLOCK.

INQUIRER: (PRESSES SEND BLOCK)

SYSTEM: ISVD HAS INFORMATION ABOUT JOBS, COLLEGES, TRADE SCHOOLS, AND MILITARY SERVICE. IN ORDER FOR YOU TO GET ANSWERS TO YOUR QUESTIONS ABOUT COLLEGES, THERE ARE A COUPLE OF GENERAL RULES TO FOLLOW. THE RULE THAT WILL PROBABLY BE MOST TROUBLESOME IS THAT TO GET INFORMATION ABOUT A COLLEGE YOU HAVE TO USE ITS CODE NUMBER OR SPELL IT OUT COMPLETELY. FOR INSTANCE, TO FIND OUT ABOUT BOSTON UNIVERSITY, YOU'D TYPE IN ITS CODE NUMBER 109 OR THE WORDS BOSTON UNIVERSITY. B. U. OR BU WOULDN'T WORK.

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: BECAUSE OF THIS INCONVENIENCE, MOST OF THE SCRIPTS WHICH SUGGEST NAMES OF JOBS OR SCHOOLS TELL YOU TO MAKE A COPY OF THE ONES YOU COME UP WITH. IF YOU DON'T HAVE A LIST WITH YOU, OR IF WE HAVE TROUBLE UNDERSTANDING YOUR REQUESTS, YOU CAN GET LISTS OF COLLEGES BY TYPING @HELP.

INQUIRER: @HELP

SYSTEM: IF YOU HAVE LEARNED HOW TO GET A NAME, YOU CAN TYPE SKIP AND THE EXPLANATION WILL NOT BE GIVEN.

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: THIS ROUTINE IS INTENDED TO PROVIDE YOU A LIST OF COLLEGES YOU CAN TYPE IN. THE MACHINE ALREADY HAS ABOUT FORTY LISTS PREPARED. YOU MAY LOOK AT THE NAMES OF THOSE LISTS AND USE ONE. YOU CAN COMBINE VARIOUS LISTS, GETTING ALL COLLEGES THAT ARE ON BOTH LISTS OR ON EITHER LIST OR ON ONE AND NOT THE OTHER. (MORE COMPLETE EXPLANATION IS GIVEN IF YOU CHOOSE TO COMBINE LISTS.) FINALLY YOU CAN CREATE YOUR OWN LIST BY STATING CRITERIA FOR SELECTING.

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: FIRST YOU MUST GET THE NAME OF A LIST (OR ITS NUMBER) AND THEN REQUEST NAMES FROM THAT LIST. NAMES WILL COME OUT IN GROUPS OF 10 WHICH YOU MAY PRINT. WHEN YOU HAVE ENOUGH, YOU CAN @STOP AND GO BACK TO THE MAIN DATA SCRIPT.

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: DO YOU WANT TO

A. CREATE

B. SELECT

C. COMBINE

LISTS OF NAMES OR

D. SEE THE COLLEGES ON A LIST YOU ALREADY HAVE.

INQUIRER: C COMBINE LISTS

SYSTEM: TO BEGIN WITH, YOUR LIST CONTAINS ALL THE ITEMS. WHAT LIST DO YOU WANT TO COMBINE YOUR ORIGINAL LIST WITH?

INQUIRER: WHAT LISTS ARE THERE?

SYSTEM: THE NAME WAS NOT RECOGNIZED. DID YOU SPELL CORRECTLY? YOU WILL KEEP CYCLING THROUGH THIS UNTIL YOU GET A CORRECT NAME OR TYPE @HELP. TRY AGAIN.

INQUIRER: @HELP

SYSTEM: THERE ARE LISTS WITH THE SCHOOLS CLASSIFIED BY SEVERAL KINDS OF FACTORS. FIRST CHOOSE A TYPE OF CLASSIFICATION FROM THE OUTLINE, THEN YOU WILL SEE ALL THE NAMES OF THOSE LISTS. YOU MUST TYPE THE EXACT NAME, SO MAKE A COPY OF THAT THAT INTEREST YOU BY PRESSING HARDCOPY. THE FACTORS ARE: TYPE THE APPROPRIATE NUMBER

1. GEOGRAPHICAL LOCATION
2. RELIGIOUS AFFILIATION
3. SEX OF STUDENT BODY
4. SIZE OF STUDENT BODY
5. AVAILABLE PROGRAMS OF STUDY
6. FINANCIAL AID
7. URBAN, SUBURBAN LOCATION
8. SPECIAL PROGRAMS
9. INTERDEPARTMENTAL PROGRAMS
10. EXTRACURRICULAR ACTIVITIES
11. RANK IN CLASS

INQUIRER: 1

SYSTEM: LOCATION

NEW ENGLAND COLLEGES
MIDDLE ATLANTIC COLLEGES
SOUTHERN COLLEGES
SOUTHWESTERN COLLEGES
WESTERN COLLEGES

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: DO YOU WANT TO:

1. SEE ANOTHER CLASS OF LISTS
2. GO BACK TO WHAT YOU WERE DOING?

INQUIRER: 2 GO BACK TO WHAT I WAS DOING

SYSTEM: WHAT LIST DO YOU WANT TO COMBINE YOUR ORIGINAL LIST WITH?

INQUIRER: NEW ENGLAND COLLEGES

SYSTEM: WHICH COMBINATION DO YOU WANT?

1. AND (THOSE THINGS ON THIS AND THE PREVIOUS COMBINED LIST)
2. OR (THOSE ON THIS LIST OR ON THE PREVIOUS COMBINED LIST)
3. NOT (THOSE ON PREVIOUS LIST BUT NOT ON THIS ONE)

INQUIRER: 1

SYSTEM: THE LISTS ARE NOW COMBINED. THE COMBINED LIST HAS 176 ITEMS IN IT. YOU MAY NOW

1. COMBINE AGAIN
2. RETURN AND HAVE YOUR LIST PRINTED
3. START OVER MAKING A NEW LIST AND LOSING THIS ONE.

TYPE 1, 2, OR 3.

INQUIRER: 2 PRINT MY LIST

SYSTEM: THE LIST WILL PRINT SEVERAL ITEMS AT A TIME, ASKING IF YOU WANT MORE. YOU SHOULD PRESS HARDCOPY FOR A PERMANENT RECORD. THIS RECORD HAS 123 COLLEGES IN IT.

CENTRAL CONNECTICUT STATE COLLEGE
DANBURY STATE COLLEGE
MANCHESTER COMMUNITY COLLEGE

SYSTEM: SOUTHERN CONNECTICUT STATE COLLEGE
UNIVERSITY OF CONNECTICUT

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: THIS RECORD HAS 123 COLLEGES IN IT. WOULD YOU LIKE TO
1. SEE THE NEXT ONES
2. SEE ANOTHER LIST
3. GET BACK TO REQUESTING DATA
TYPE THE NUMBER OF YOUR CHOICE

INQUIRER: 3

SYSTEM: AFTER YOU TYPE IN THE NUMBER OR FULL NAME OF THE COLLEGE YOU'RE INTERESTED IN, WE'LL ASK WHAT YOU WANT TO KNOW ABOUT IT. AGAIN, IF YOU'RE NOT SURE WHAT SORT OF INFORMATION YOU WANT TO KNOW OR IF WE HAVE TROUBLE FIGURING OUT YOUR QUESTIONS ABOUT THE COLLEGE, TYPE @HELP.

INQUIRER: UNIVERSITY OF CONNECTICUT

SYSTEM: WHAT DO YOU WANT TO KNOW ABOUT UNIVERSITY OF CONNECTICUT?

INQUIRER: LOCATION

SYSTEM: COLLEGE = UNIVERSITY OF CONNECTICUT
VARIABLE = LOCATION
UNIVERSITY OF CONNECTICUT IS LOCATED IN STORRS, CONNECTICUT.

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: TYPE IN THE NAME OF THE COLLEGE YOU WANT INFORMATION ABOUT OR THE IDENTIFICATION NUMBER. JUST PUSH SEND BLOCK IF YOU WANT TO KNOW SOMETHING ELSE ABOUT THE SAME COLLEGE.

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: WHAT DO YOU WANT TO KNOW ABOUT UNIVERSITY OF CONNECTICUT?

INQUIRER: COST

SYSTEM: COLLEGE = UNIVERSITY OF CONNECTICUT
VARIABLE = COST
THE TUITION AT UNIVERSITY OF CONNECTICUT IS $190 FOR STATE RESIDENTS AND $400 FOR NON RESIDENTS.

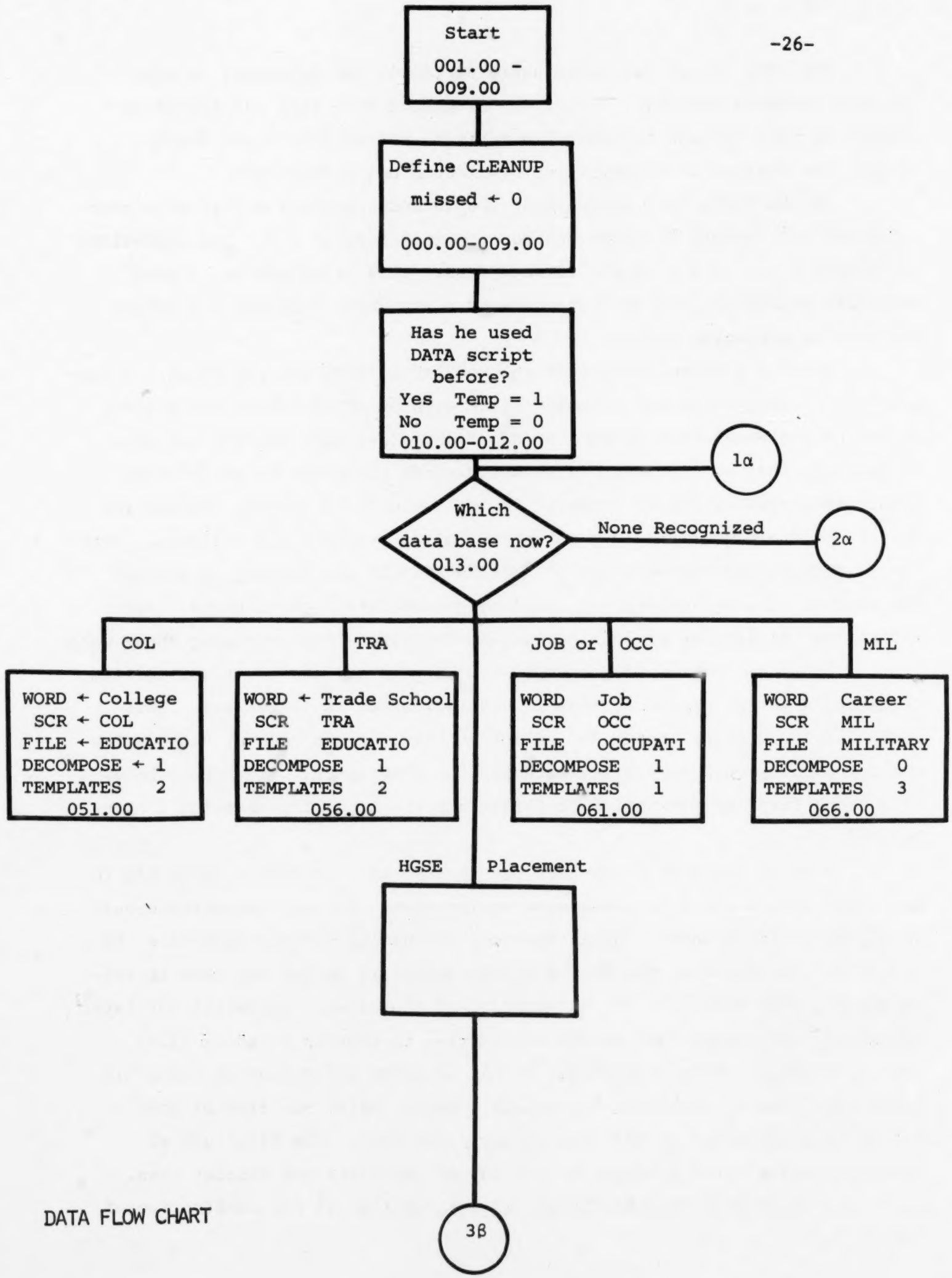
The DATA script was notoriously difficult for inquirers to use. Two main problems emerged. First the directions were long and forced inquirers to wade through interminable material before they could begin. Second, the variety of capabilities bewildered naive inquirers.

As the field test progressed, the introduction was drastically shortened and the variety of functions was not mentioned at all. The capability still exists but is not taught to inquirers. This procedure was a good temporary expedient, but is inadequate as a premanent solution. A better solution is presented later.

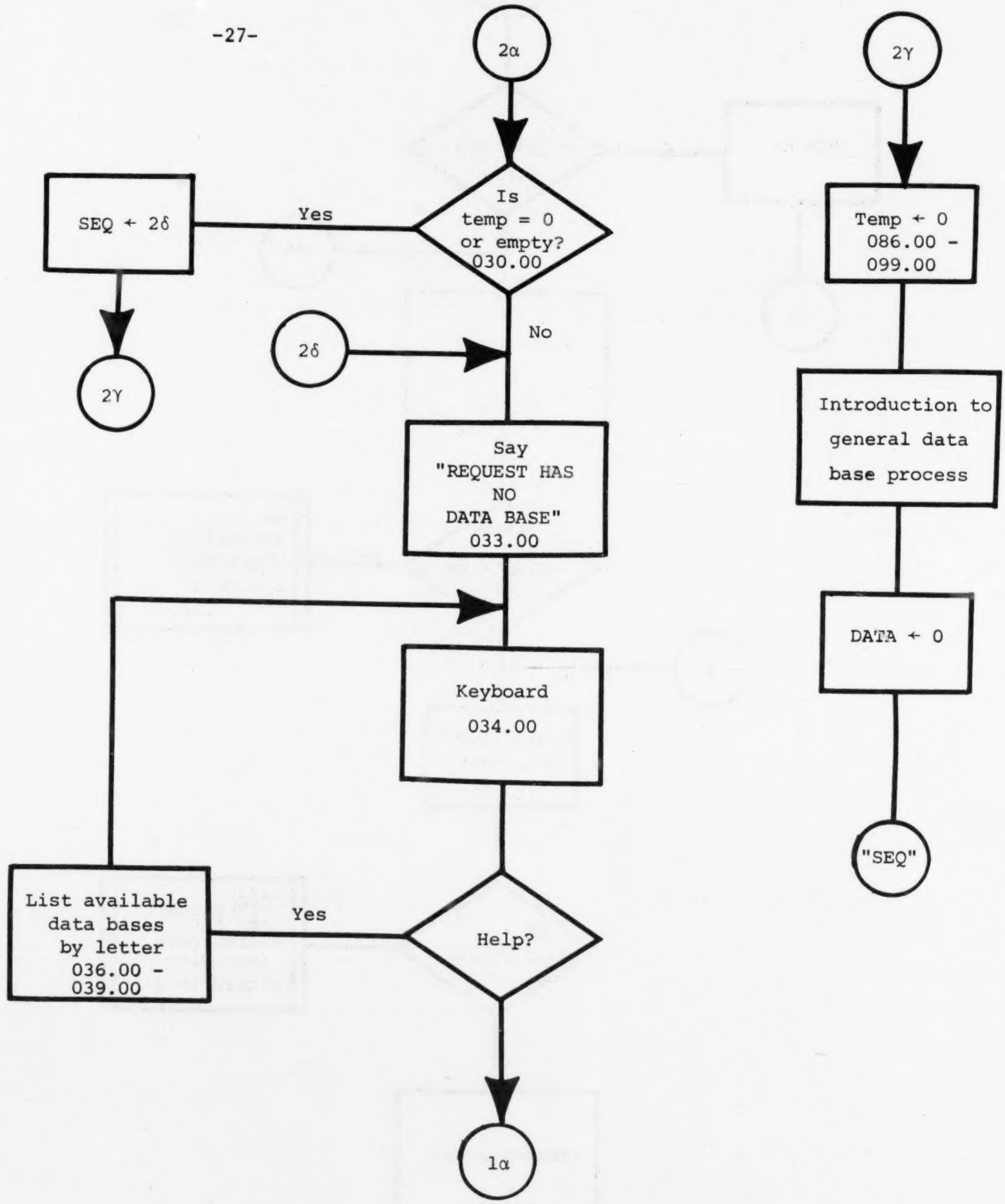
Another problem arose from the VARSIMPLE#(SCR) scripts which are supposed to recognize natural language statements about variables and provide a list of sentences that answer the question. They were written too specifically so that an inquirer had to use exactly the words we anticipated rather than those words or synonyms in any sentence or phrase. T ward the end of the field test this was corrected for occupations and colleges. Statistics summarizing the workings of VARSIMPLE#(SCR) and helping to develop the natural language capability could be gathered from the logtape. Specifications for logtape processing include directions for obtaining these data.

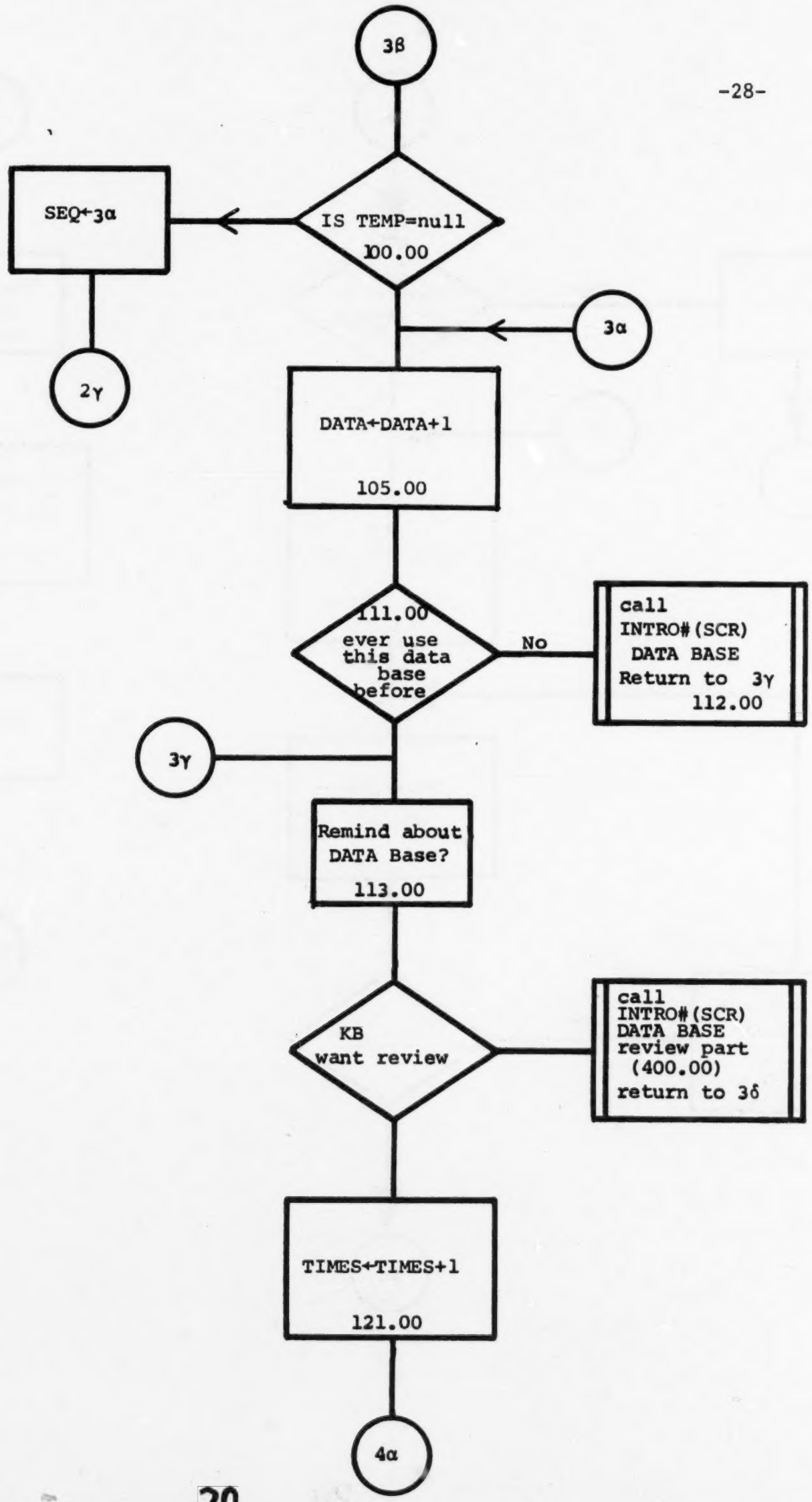
The DATA routine offers an opportunity to explore the possibilities of making a script appear to grow in sophistication as it is used. This would involve testing the amount of use an inquirer has made of the routine and introducing more complexity with greater experience. The scheme below is a crude first approximation to illustrate the kind of processing I have in mind.

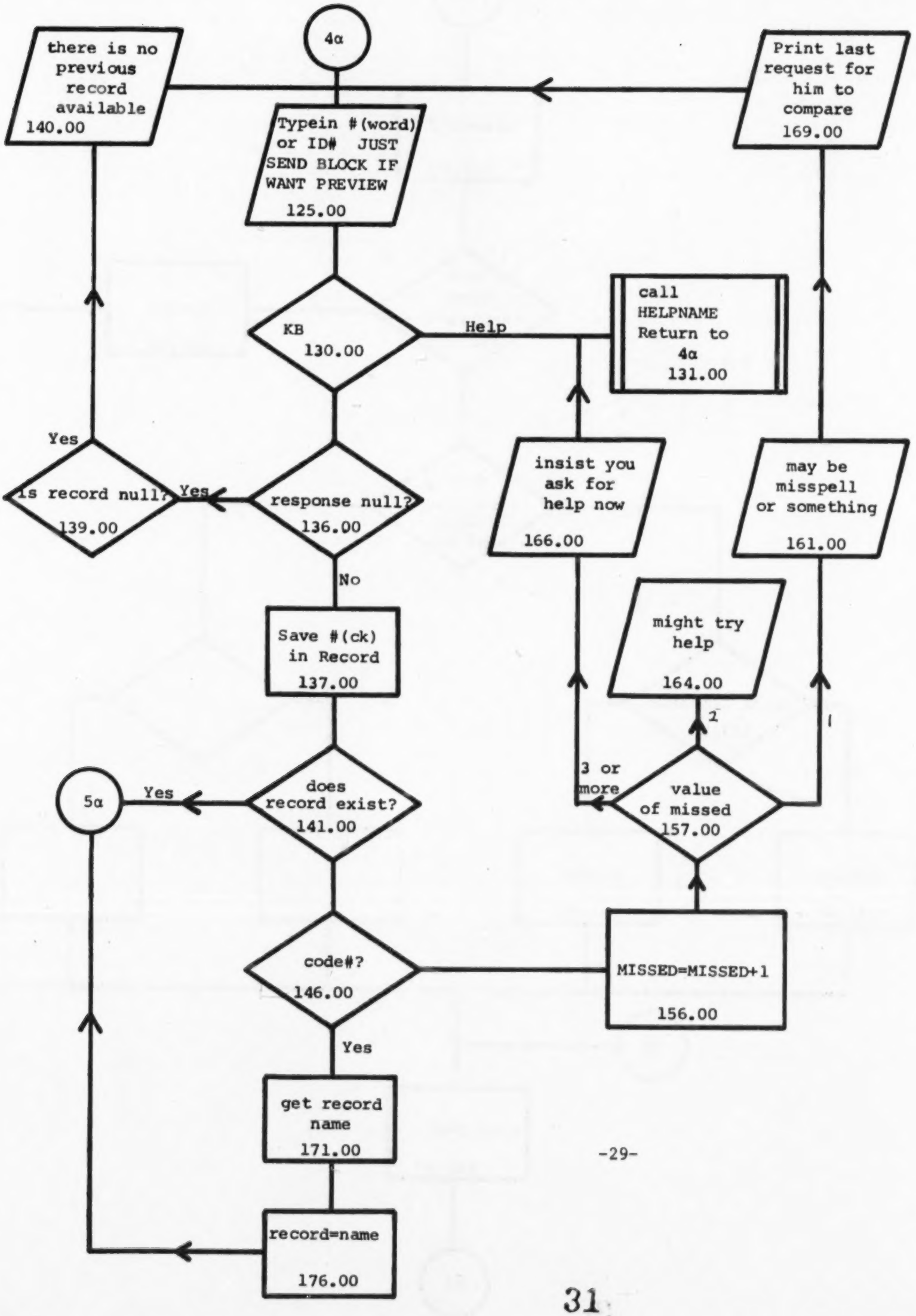
When an inquirer first uses the DATA script, it should force him to use items from a previous preference script list. He will concentrate only on typing variable names. After successfully naming several variables, he should be introduced to the HELPVAR#(SCR) script if he has not used it voluntarily. This will give him an overview of the allowed variables for later reference. Following this he should be asked to type in a school (job) name by himself. When successful, he can be given the option of using his preference list to automatically supply a name. Later his list of good or bad items accumulated in GOBI could supply the item. The first use of HELPNAME should allow inquirer to pck any of our lists and display them. With more experience in DATA, he should use the logical and combination of

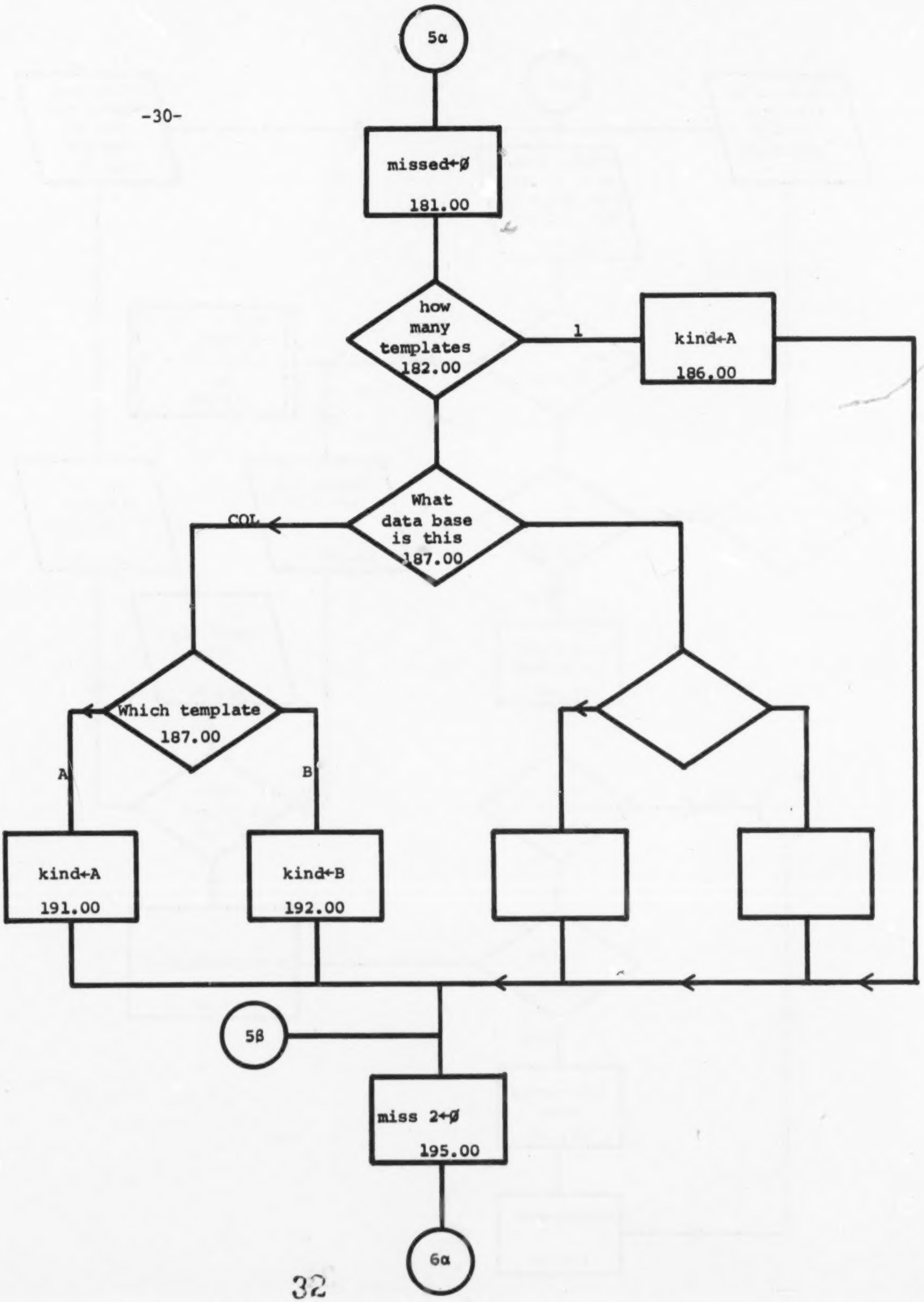


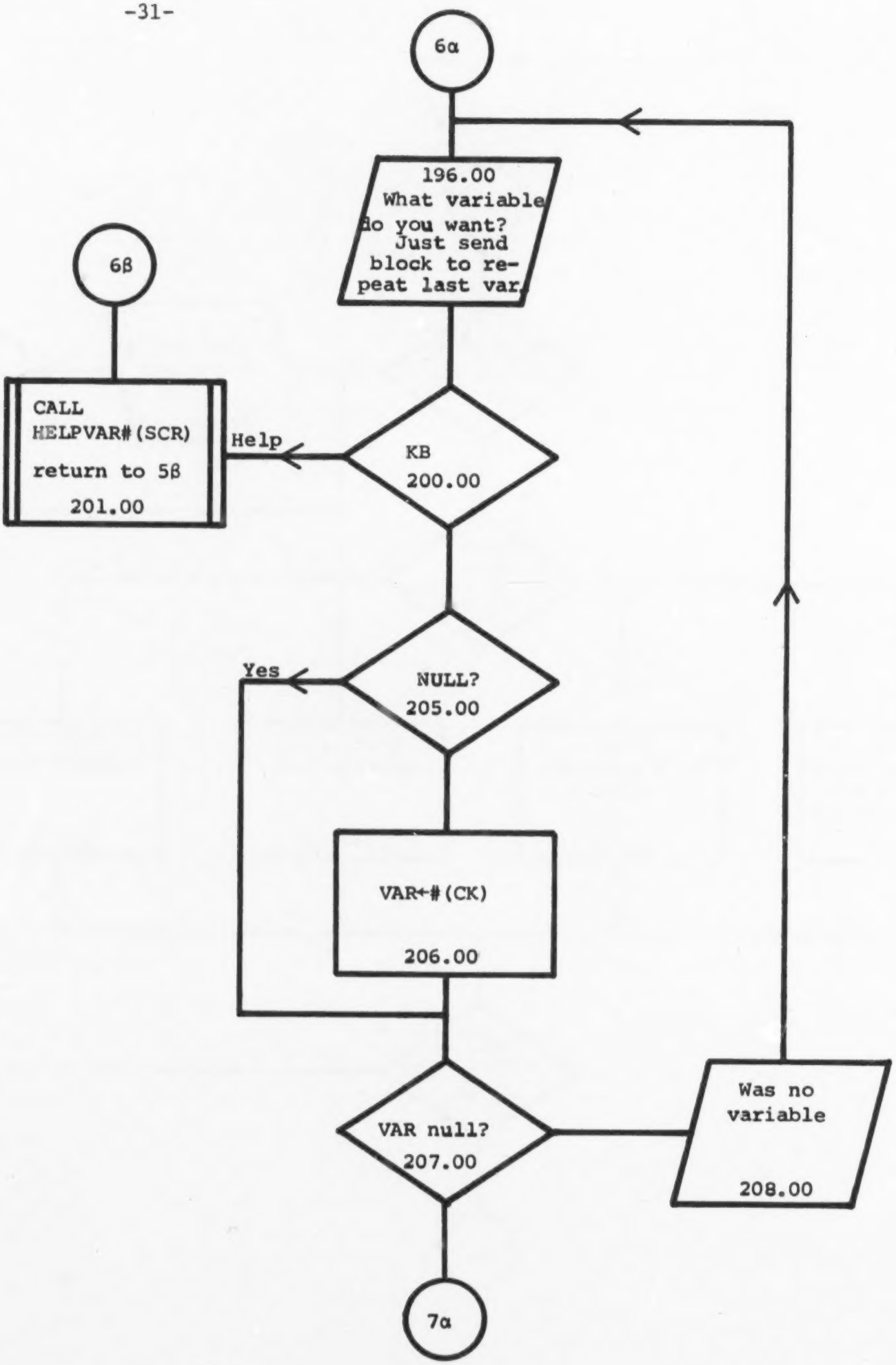
DATA FLOW CHART

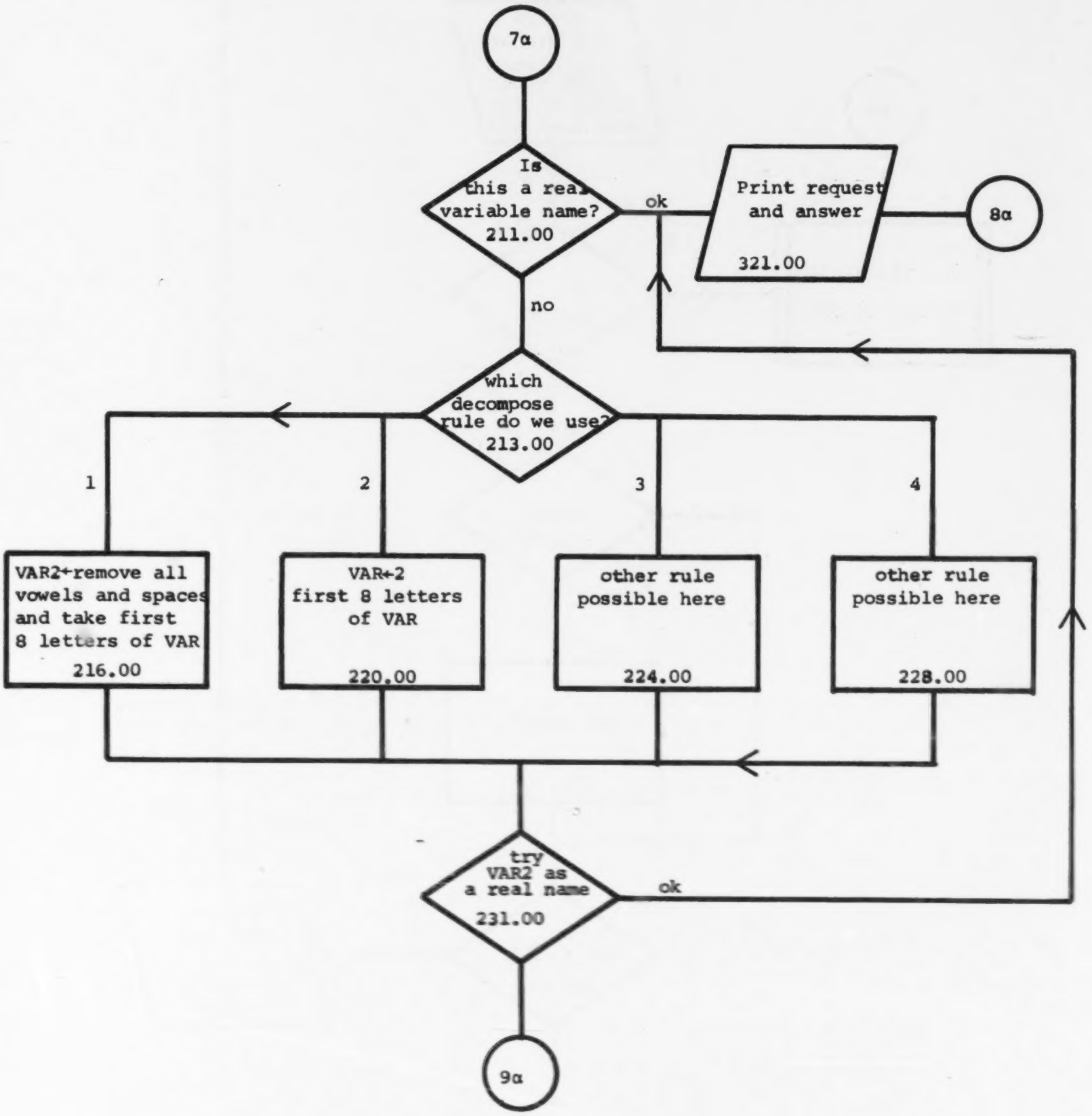

2α
SEQ ← 2δ
Yes
Is
temp = 0
or empty?
030.00
No
2δ
2γ
Say
"REQUEST HAS
NO
DATA BASE"
033.00
Keyboard
034.00
List available
data bases
by letter
036.00 -
039.00
Yes
Help?
1α
2γ
Temp ← 0
086.00 -
099.00
Introduction to
general data
base process
DATA ← 0
"SEQ"

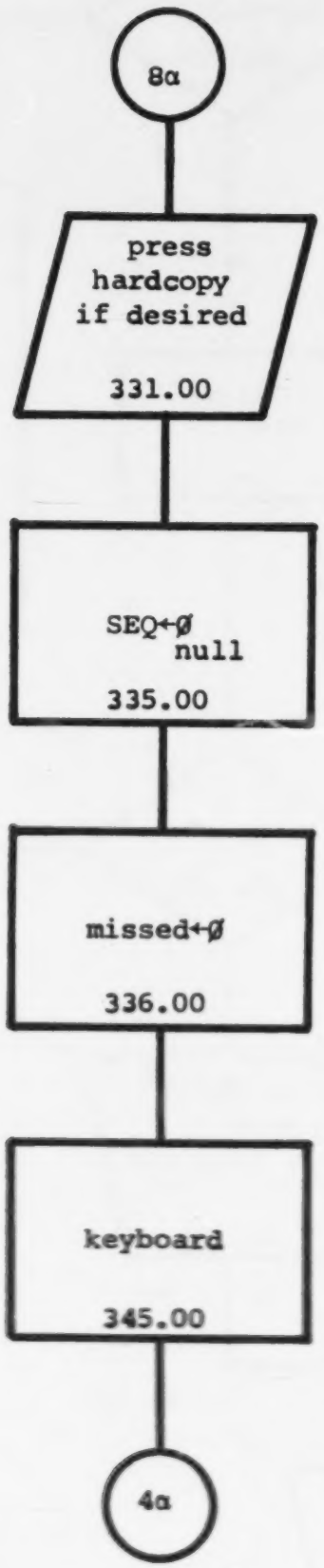

3β
SEQ←3α
IS TEMP=null
100.00
3α
2γ
DATA←DATA+1
105.00
111.00
ever use
this data
base
before
No
call
INTRO#(SCR)
DATA BASE
Return to 3γ
112.00
3γ
Remind about
DATA Base?
113.00
KB
want review
call
INTRO#(SCR)
DATA BASE
review part
(400.00)
return to 3δ
TIMES←TIMES+1
121.00
4α

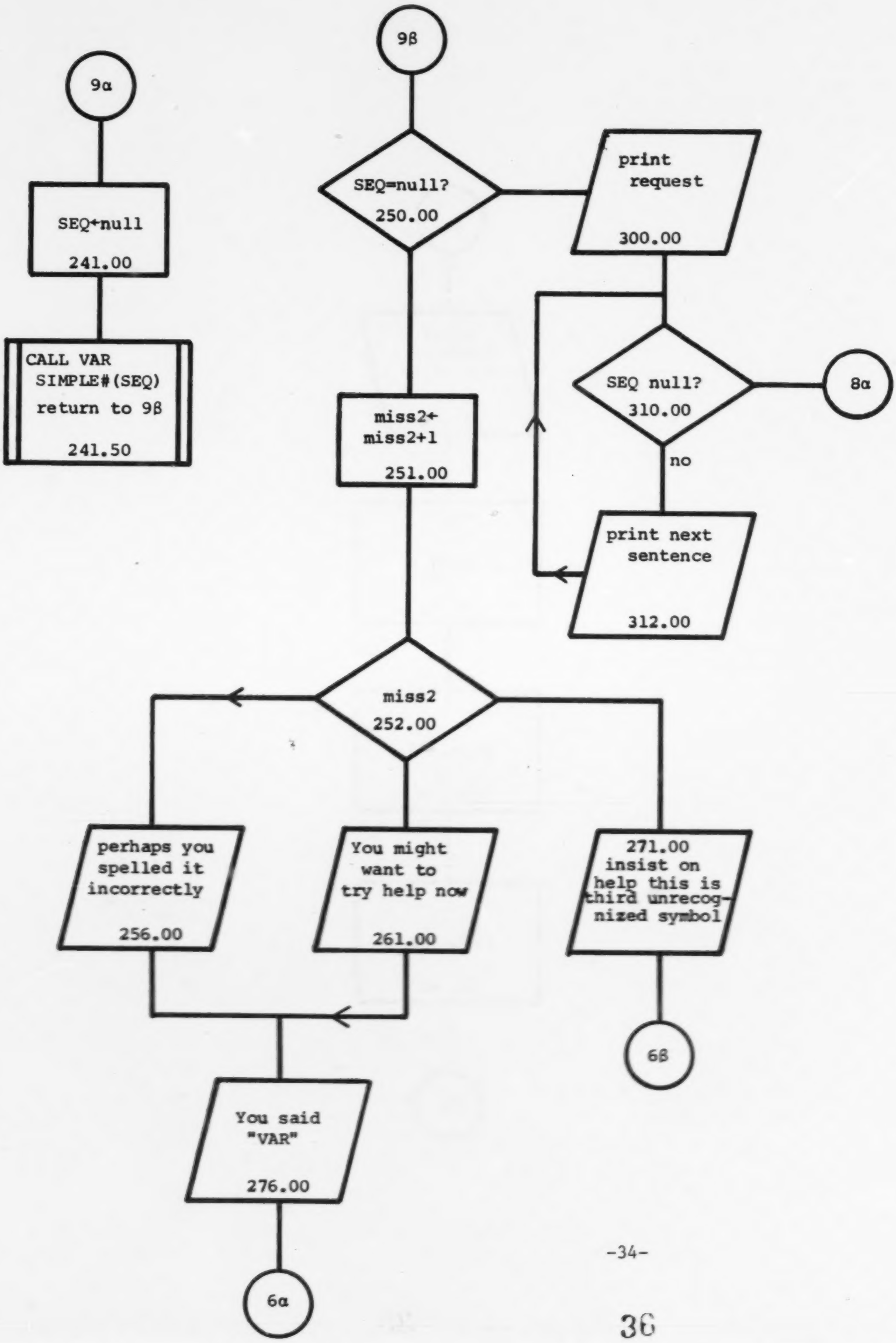

4α
there is no previous record available 140.00
Print last request for him to compare 169.00
Typein #(word) or ID# JUST SEND BLOCK IF WANT PREVIEW 125.00
KB 130.00
Help
call HELPNAME Return to 4α 131.00
Yes
is record null? 139.00
Yes
response null? 136.00
insist you ask for help now 166.00
may be misspell or something 161.00
No
Save #(ck) in Record 137.00
might try help 164.00
2
1
3 or more
value of missed 157.00
5α
Yes
does record exist? 141.00
code#? 146.00
MISSED=MISSED+1 156.00
Yes
get record name 171.00
record=name 176.00

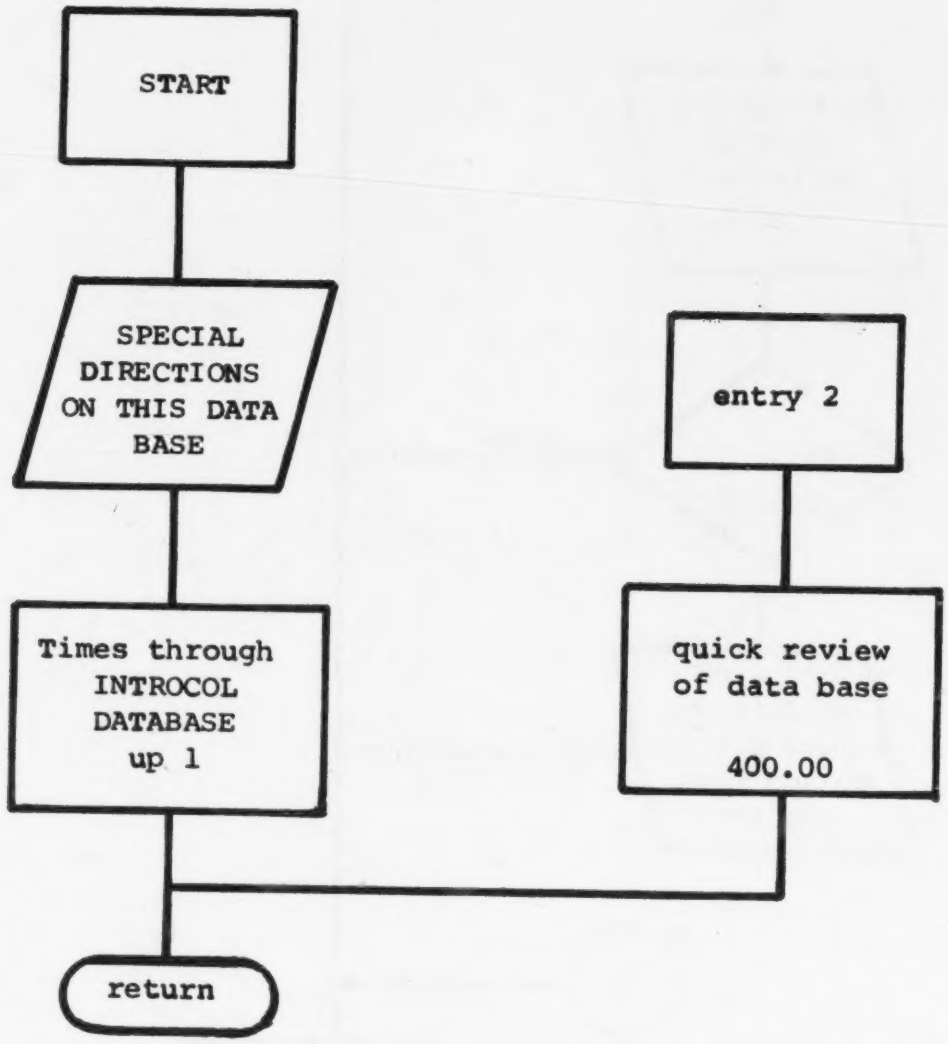

5α
missed←ø
181.00
how many templates
182.00
1
kind←A
186.00
What data base is this
187.00
COL
Which template
187.00
A
B
kind←A
191.00
kind←B
192.00
5β
miss 2←ø
195.00
6α


6α
196.00
What variable do you want? Just send block to repeat last var.
6β
CALL HELPVAR#(SCR)
return to 5β
201.00
Help
KB
200.00
Yes
NULL?
205.00
VAR←#(CK)
206.00
VAR null?
207.00
Was no variable
208.00
7α


7α
Is this a real variable name? 211.00
ok
Print request and answer
321.00
8α
no
which decompose rule do we use? 213.00
1
2
3
4
VAR2←remove all vowels and spaces and take first 8 letters of VAR
216.00
VAR←2 first 8 letters of VAR
220.00
other rule possible here
224.00
other rule possible here
228.00
try VAR2 as a real name
231.00
ok
9α

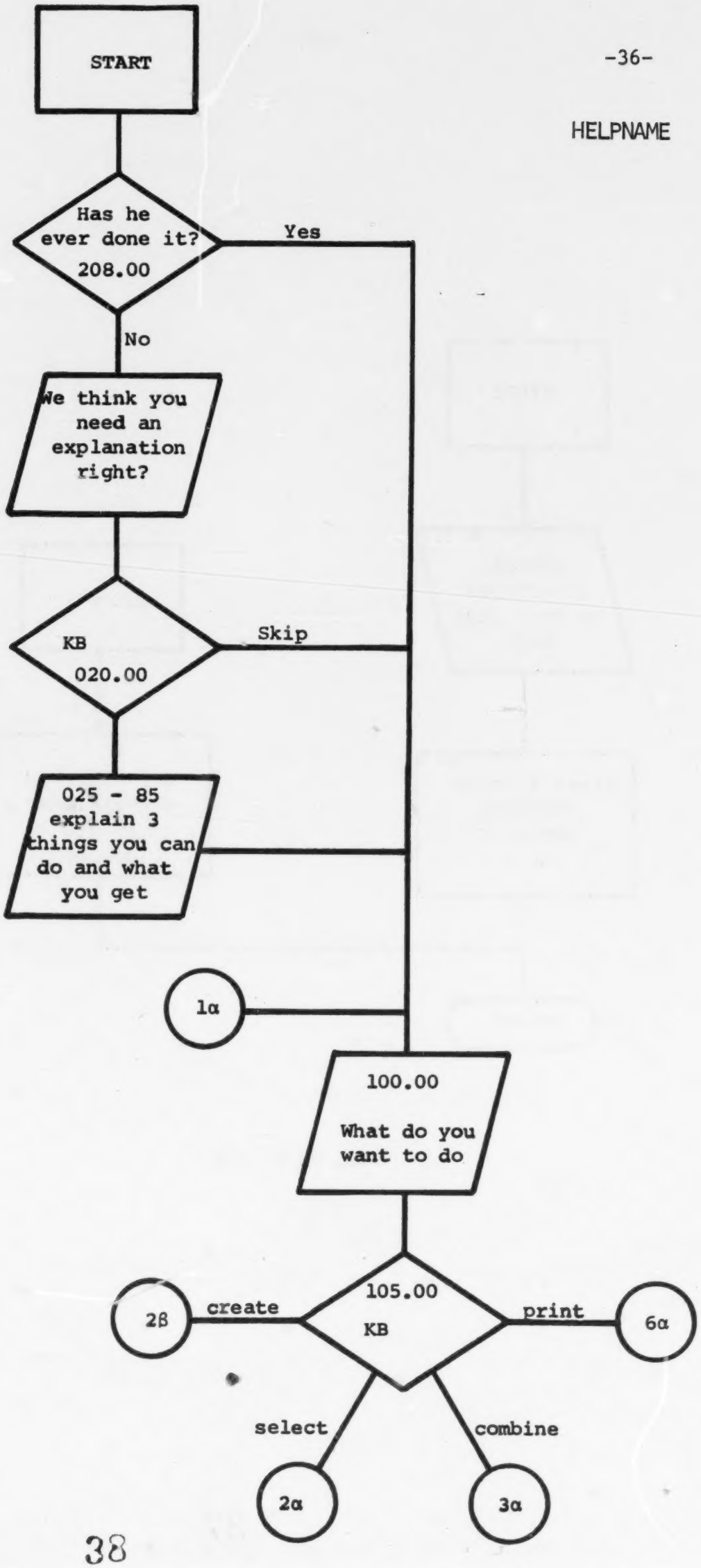

8α
press
hardcopy
if desired
331.00
SEQ←Ø
null
335.00
missed←Ø
336.00
keyboard
345.00
4α

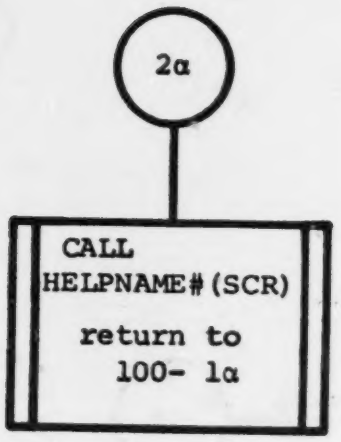

9α
SEQ←null
241.00
CALL VAR
SIMPLE#(SEQ)
return to 9β
241.50
9β
SEQ=null?
250.00
print
request
300.00
miss2←
miss2+1
251.00
SEQ null?
310.00
8α
no
print next
sentence
312.00
miss2
252.00
perhaps you
spelled it
incorrectly
256.00
You might
want to
try help now
261.00
271.00
insist on
help this is
third unrecog-
nized symbol
6β
You said
"VAR"
276.00
6α



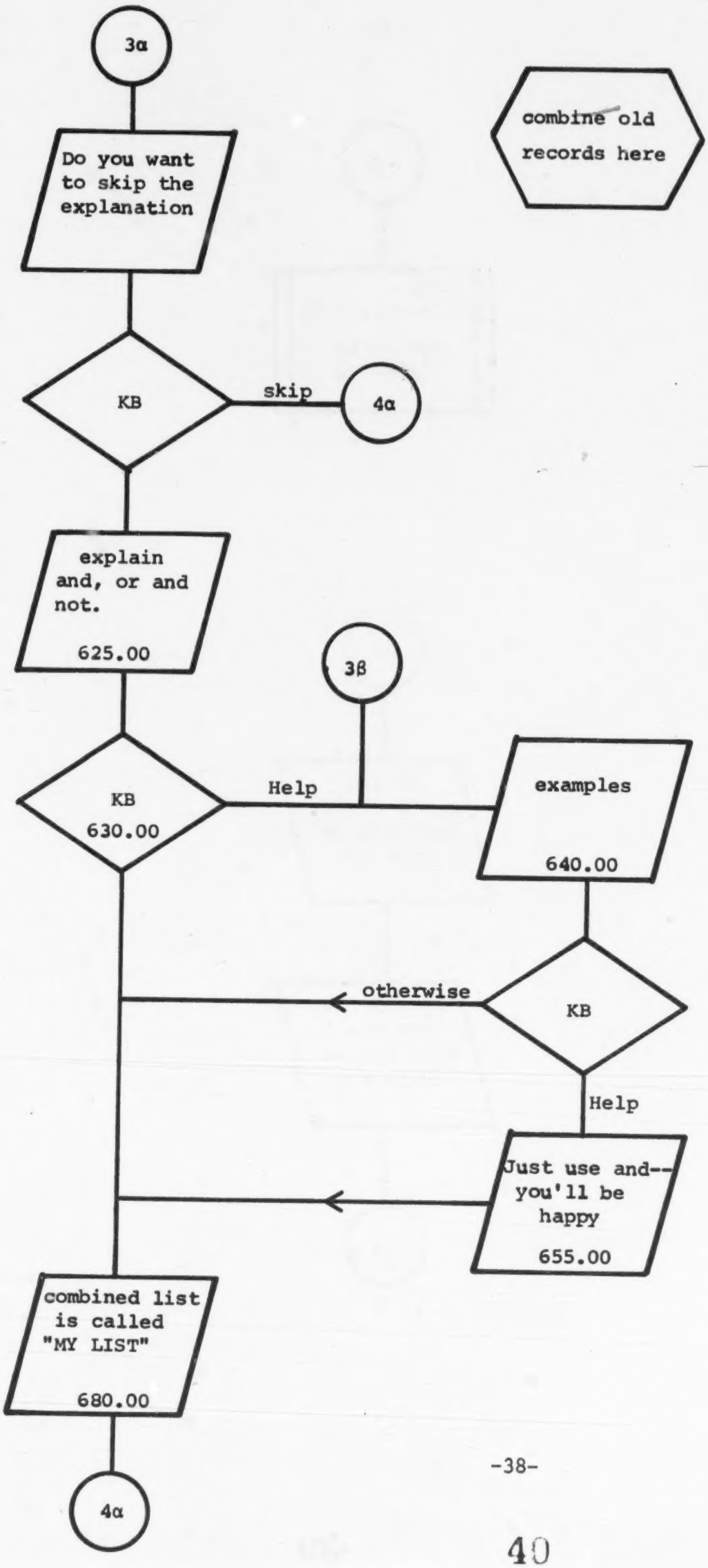
INTROCOL DATA BASE

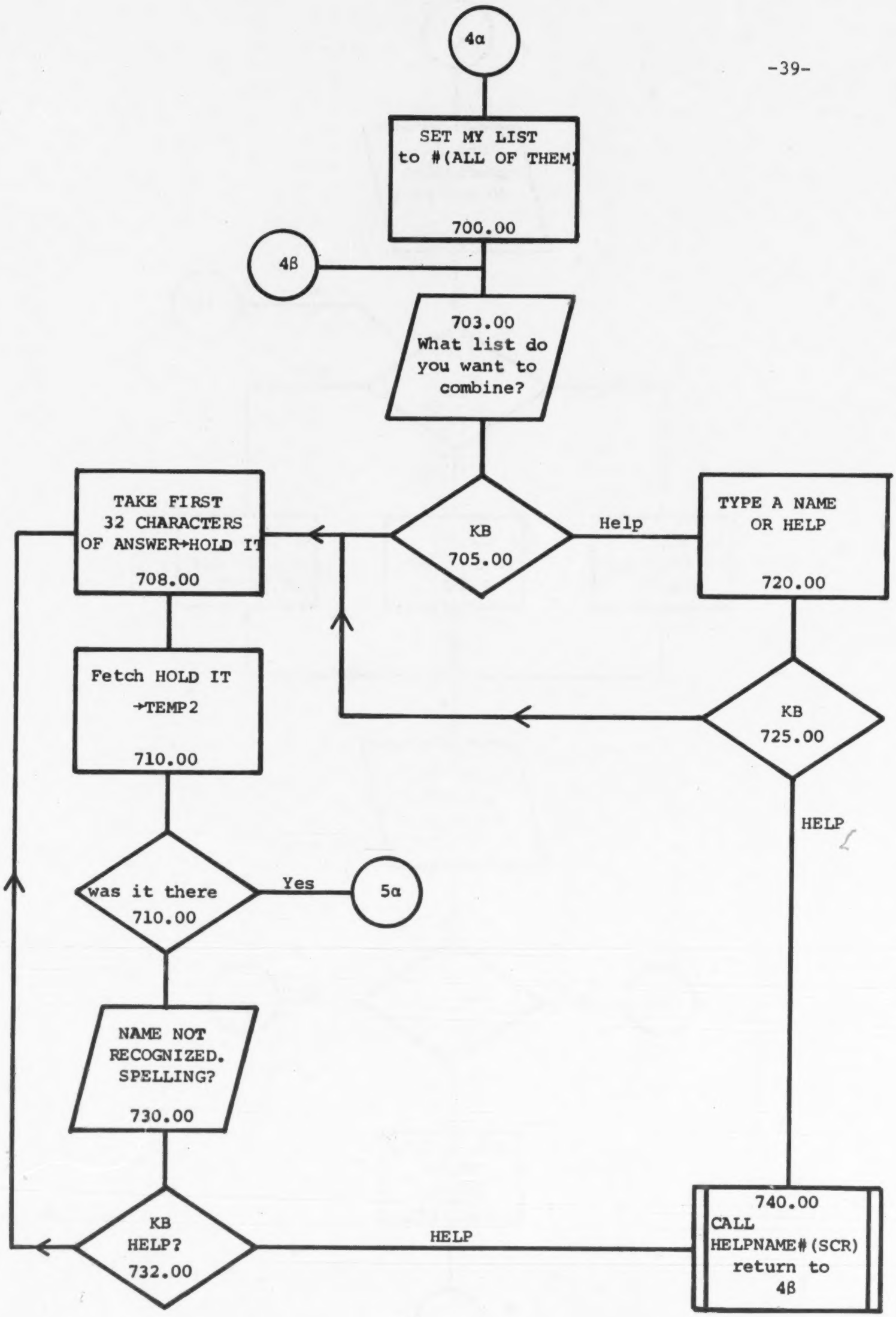
HELPNAME

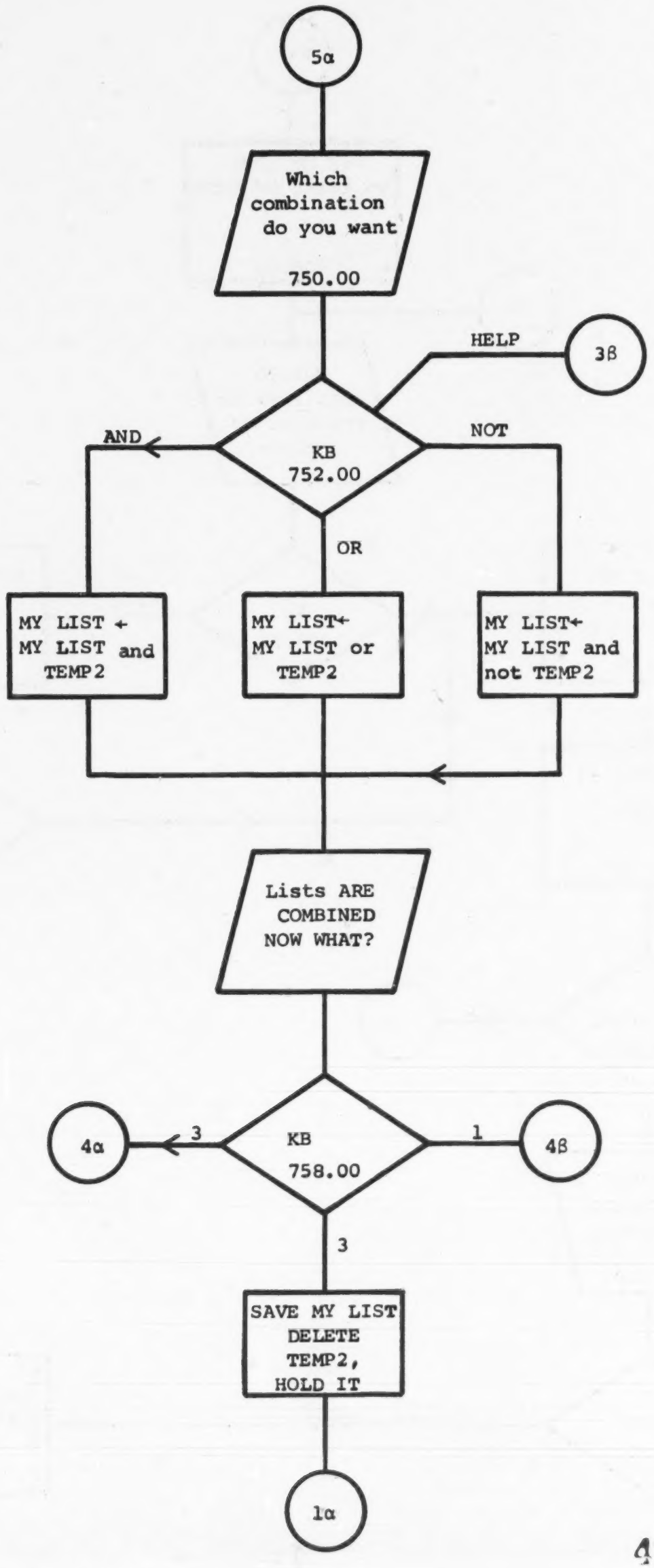

START
Has he ever done it?
208.00
Yes
No
We think you need an explanation right?
KB
020.00
Skip
025 - 85
explain 3 things you can do and what you get
1α
100.00
What do you want to do
105.00
KB
2β
create
print
6α
select
combine
2α
3α

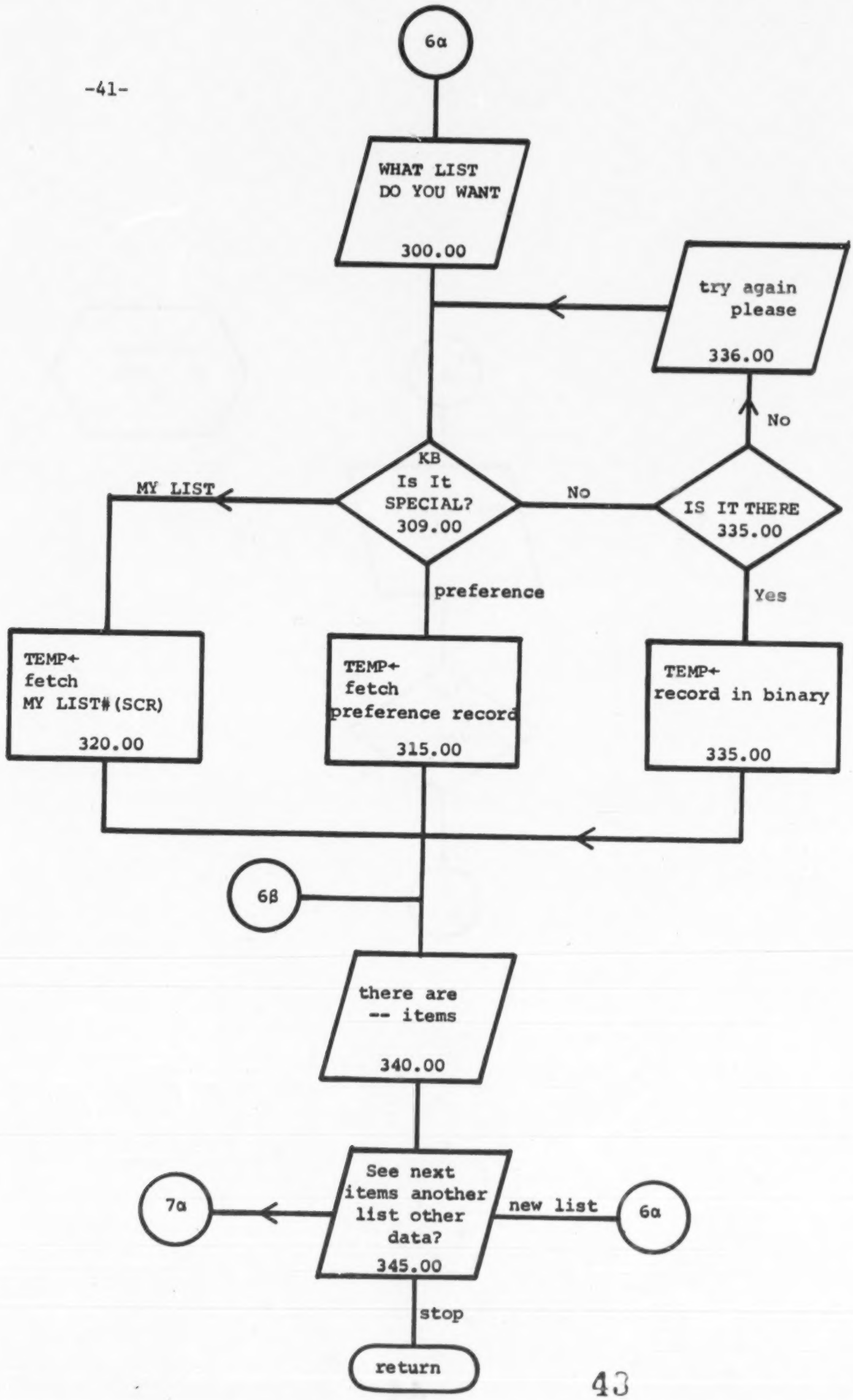

2α
CALL
HELPNAME#(SCR)
return to
100- 1α

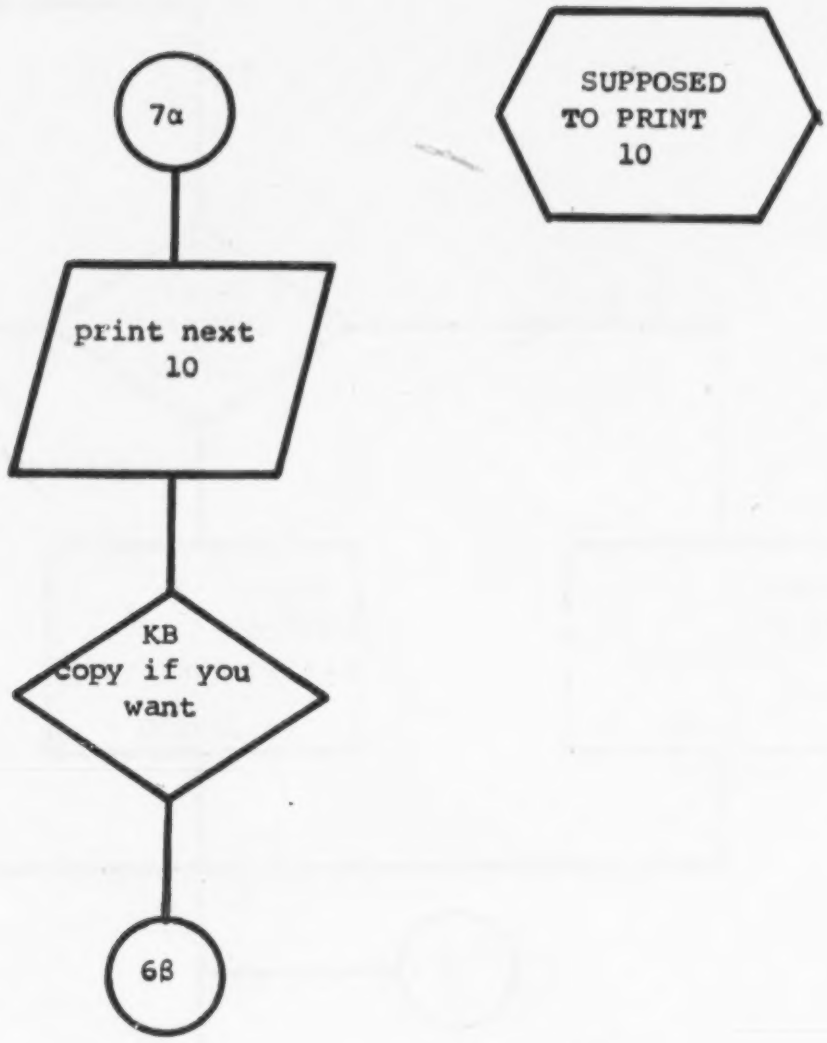
2β
800.00
can't create
records on
line now
Specify the
one you
want and we
will try
1α

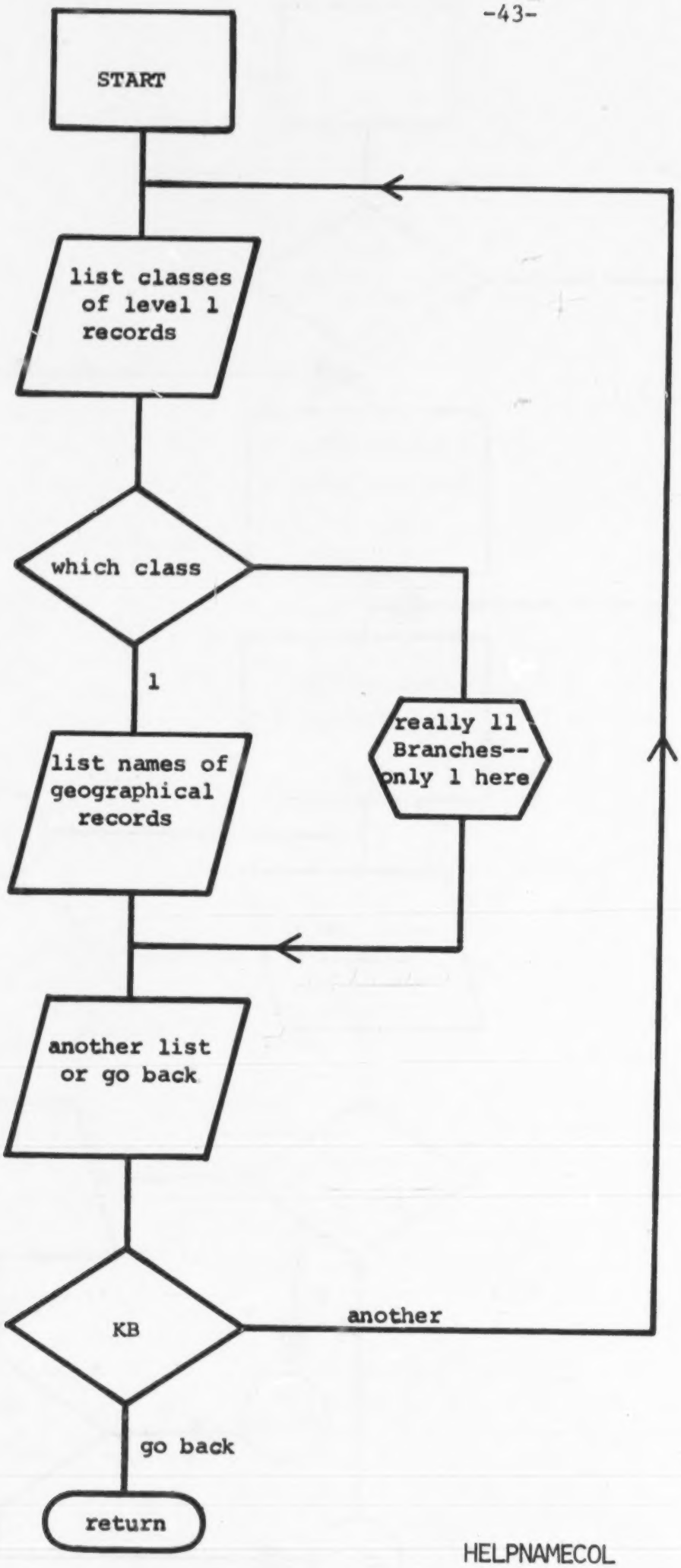

3α
Do you want to skip the explanation
KB
skip
4α
explain and, or and not.
625.00
KB
630.00
Help
3β
examples
640.00
otherwise
KB
Help
Just use and-- you'll be happy
655.00
combined list is called "MY LIST"
680.00
4α
combine old records here


4α
SET MY LIST
to #(ALL OF THEM)
700.00
4β
703.00
What list do
you want to
combine?
TAKE FIRST
32 CHARACTERS
OF ANSWER→HOLD IT
708.00
KB
705.00
Help
TYPE A NAME
OR HELP
720.00
Fetch HOLD IT
→TEMP2
710.00
KB
725.00
HELP
was it there
710.00
Yes
5α
NAME NOT
RECOGNIZED.
SPELLING?
730.00
KB
HELP?
732.00
HELP
740.00
CALL
HELPNAME#(SCR)
return to
4β

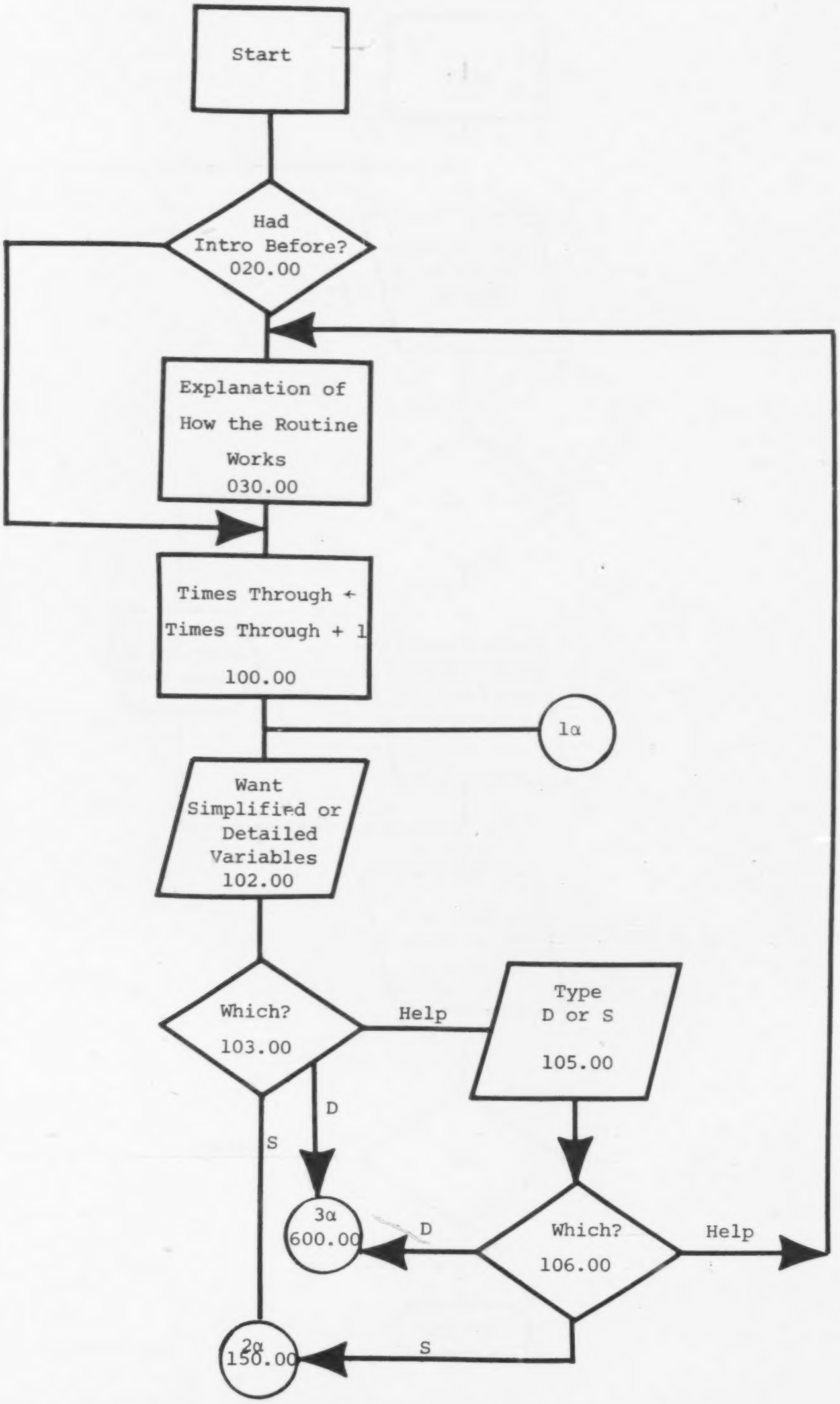

5α
Which combination do you want
750.00
HELP
3β
AND
KB
752.00
NOT
OR
MY LIST ← MY LIST and TEMP2
MY LIST← MY LIST or TEMP2
MY LIST← MY LIST and not TEMP2
Lists ARE COMBINED NOW WHAT?
4α
3
KB
758.00
1
4β
3
SAVE MY LIST DELETE TEMP2, HOLD IT
1α


6α
WHAT LIST
DO YOU WANT
300.00
try again
please
336.00
No
KB
Is It
SPECIAL?
309.00
MY LIST
No
IS IT THERE
335.00
preference
Yes
TEMP←
fetch
MY LIST#(SCR)
320.00
TEMP←
fetch
preference record
315.00
TEMP←
record in binary
335.00
6β
there are
-- items
340.00
See next
items another
list other
data?
345.00
7α
new list
6α
stop
return

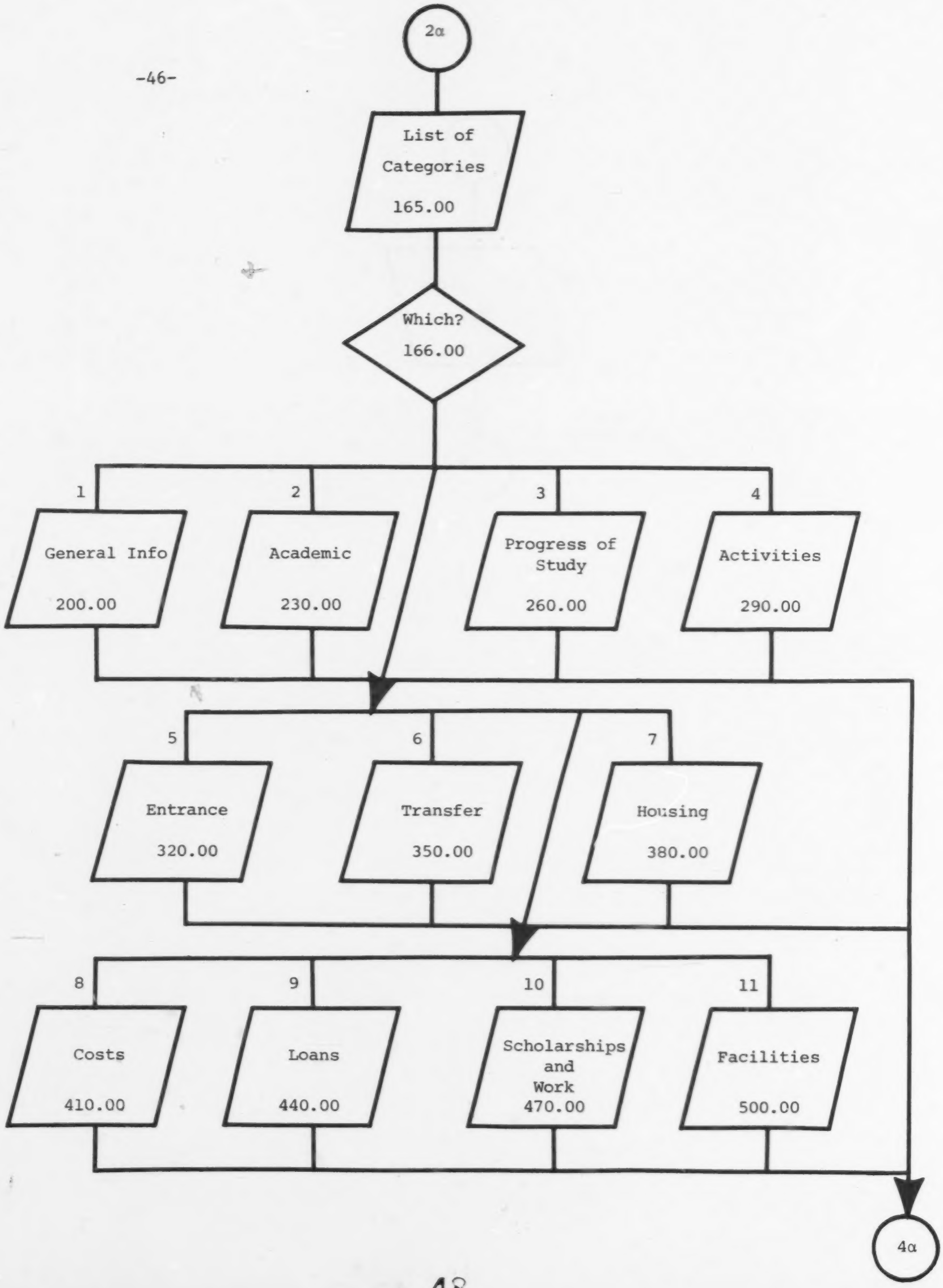

7α
print next
10
KB
copy if you
want
6β
SUPPOSED
TO PRINT
10

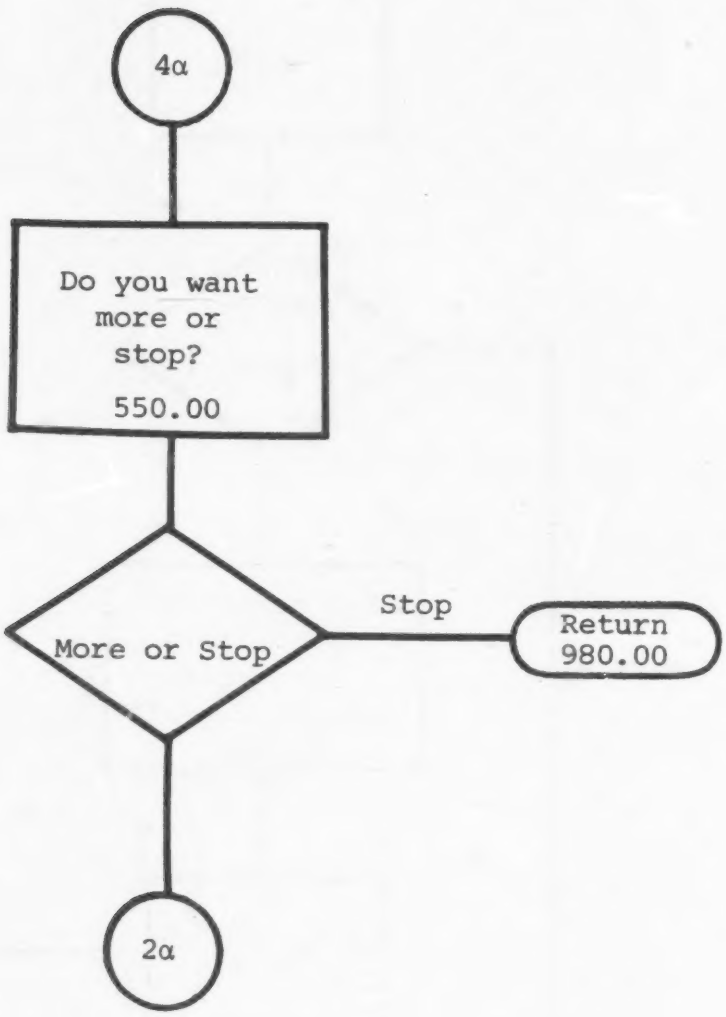


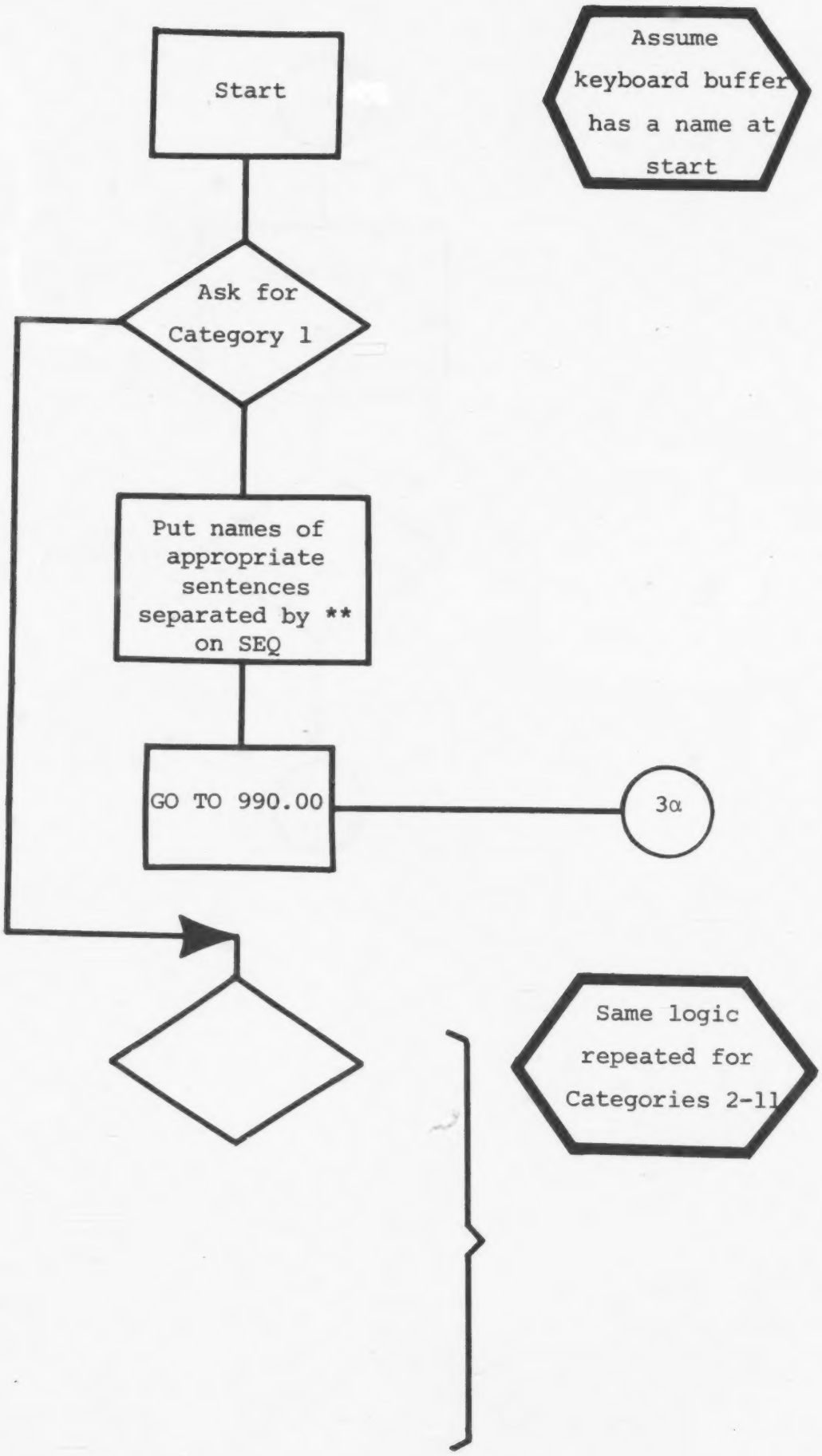
HELPNAMECOL


Start
Had Intro Before? 020.00
Explanation of How the Routine Works 030.00
Times Through ← Times Through + 1 100.00
1α
Want Simplified or Detailed Variables 102.00
Which? 103.00
Help
Type D or S 105.00
D
S
3α 600.00
D
Which? 106.00
Help
2α 150.00
S

3α
Sorry can't
do it yet
600.00
1α

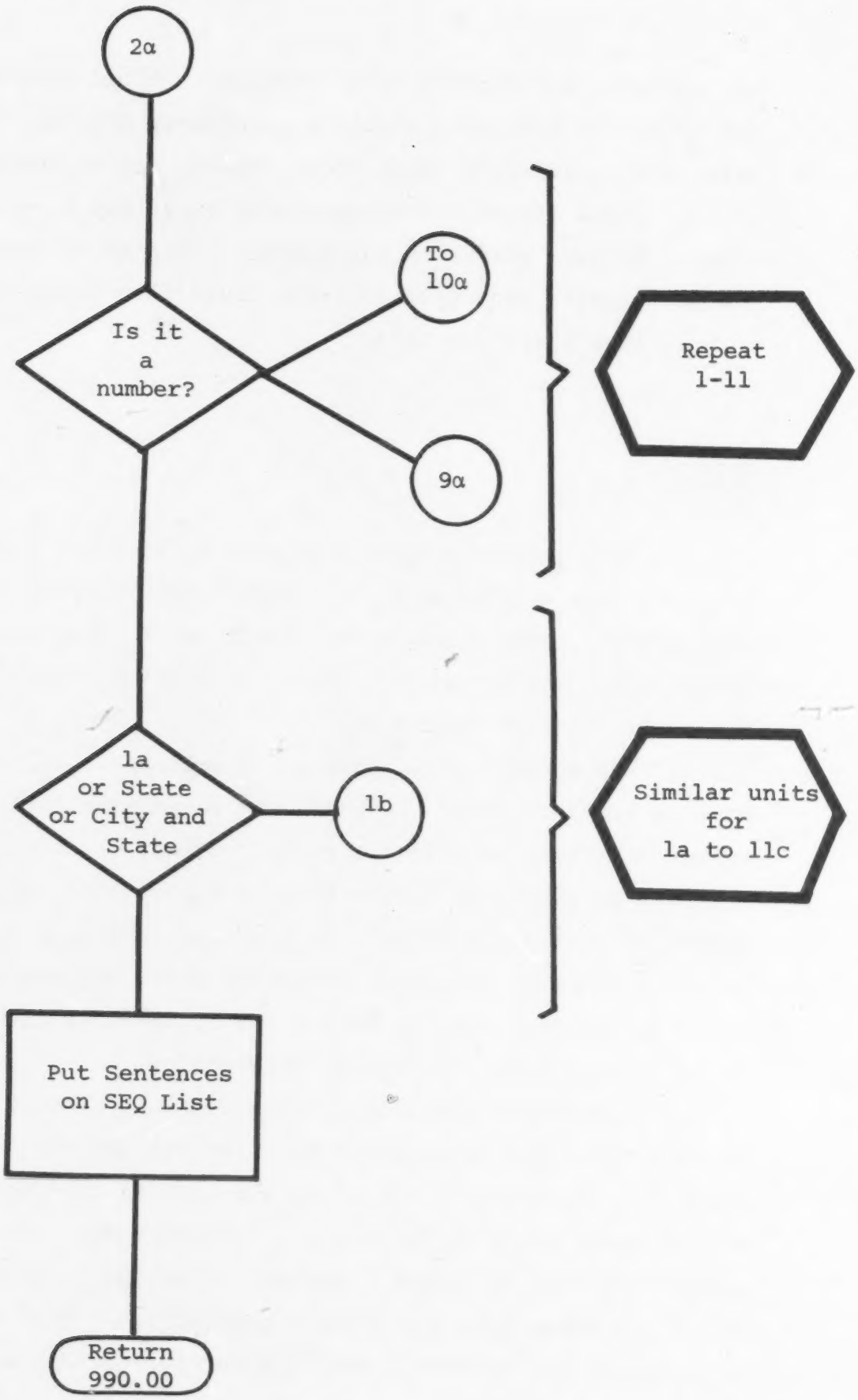

2α
List of Categories
165.00
Which?
166.00
1
General Info
200.00
2
Academic
230.00
3
Progress of Study
260.00
4
Activities
290.00
5
Entrance
320.00
6
Transfer
350.00
7
Housing
380.00
8
Costs
410.00
9
Loans
440.00
10
Scholarships and Work
470.00
11
Facilities
500.00
4α


4α
Do you want more or stop?
550.00
More or Stop
Stop
Return 980.00
2α


Start
Assume keyboard buffer has a name at start
Ask for Category 1
Put names of appropriate sentences separated by ** on SEQ
GO TO 990.00
3α
Same logic repeated for Categories 2-11

VARSIMPLECOL

2α


2α
To
10α
Is it
a
number?
9α
Repeat
1-11
1a
or State
or City and
State
1b
Similar units
for
1a to 11c
Put Sentences
on SEQ List
Return
990.00

two vectors, and explore their contents. After several uses of this facility the choice of combining should be routinely offered, but not insisted upon. Later still, he could learn about logical and/or negation.

Other criteria than cumulated usage may be employed in this decision logic. Perhaps grades in mathematics would be an appropriate discrimination. Some internally generated criteria could also serve.

[see Chart 7 - DATA]

QUIT

This script is used only when an inquirer types @QUIT. The current function is simply to store the STAKEN vector, compute and store ETAKEN, store DBVEC, compute and store EDBVEC and to list the names of data bases referenced. The script terminates by calling logout (LO) and returning to the script network beginning.

The script works rapidly and well now. All the essential information is stored. There has been some suggestion that some question and answer material, including a verbal summary of the day's interaction be included in QUIT. In theory this is a good idea, getting the inquirer to assess his current feelings; in practice, the time involved would cause severe problems. Inquirers typically @QUIT only seconds before the bell for a new period rings or before the system goes down for the day. There is no time to spend in further interaction.

A possible solution to this conflict between theory and capability would require system programming to access the clock. Even then some mechanism for inquiring of the clock and knowing the length of time an inquirer has to spend would be necessary. Different inquirers will take different amounts of time to respond, and will work for varying lengths of time. All in all it seems like a difficult proposition. Much the same function could be accomplished in STOP. See STOP for further discussion.

STOP

STOP is really two scripts. One begins at step 473.2Ø and handles errors in script execution. When a step is not found, GLURP automatically passes control to this step.

NO SCRIPT NETWORK WILL WORK WITHOUT THIS SCRIPT STEP.
Control is passed on with the LINK list. If no scripts remain on the list, control is passed to ORIENTI Ø5Ø.ØØ. An improvement would be to log the value of IOERR to the tape for further reference.

The other script is called when inquirer types @STOP. It handles communication through the STOP list. At the same time it cleans up the LINK list so that links to the STOPPED script and beyond are eliminated.
Note I believe the current version of Ø4Ø.ØØ has caused trouble insidiously. It should read #(PT:LINK). It now does its duty twice instead of once. Check this through the logtape.
Note The fact that STOP logs the names of all forms currently in memory gives a possible check on which scripts are not deleting things or are using CLEANUP improperly.

In the STOP script you could take the opportunity to ask the inquirer why he stopped. Offer him a multiple choice set and the option of a natural language response. This would provide concrete feedback on the feelings generated by interaction with specific scripts. I believe this is more desirable than the overview of feelings provided by a similar question in @QUIT.

SUMMARY

The SUMMARY script is designed to give a quick review of important events of the previous interaction or the current interaction. It could logically be extended to any interaction since it depends only on N, the times through the system for its analysis. The current value of STAKEN should be available to review the session. The values of STAKEN#(N) will be fetched for earlier interactions.

In the current implementation, STAKEN is stored whenever ORIENTI is executed, so some data will always be available. As explained in INTRODUCTIONTOSYSTEM, inquirers can skip incrementing the value of TIMES THROUGH:SYSTEM. This will lead to erroneous SUMMARY. The changes suggested in ITS should alleviate this problem.

Steps Ø15.ØØ - Ø17.ØØ turn off bits corresponding to scripts which are supposed to be invisible to the inquirer. See Table 13 of INQTABLE and Specifications of Level One Vectors for INQTABLE for further specification. Steps Ø17.2Ø - Ø17.25 turn off bits numbered over 236, since they are not included in the decode tables on line and cause a Z branch for a decode table error. The tables 13, 14, 15 and the level one records in INQTABLE should be extended beyond 236 and steps Ø17.2Ø - Ø17.25 deleted from the script.

To fully understand the undocumented coding in SUMMARY requires the use of STORAGE IN ISVD to describe the meaning of the references. There are numerous errors of communication reflected in SUMMARY as well as some subtle coding errors. Together they combine to make the script fail to operate 75% of the time.

No one has made detailed tests of conditions or controlled for various contamination (like failing to @QUIT at the last session). It is therefore impossible to say what changes are required technically. The logtape provides a record of action, and together with STORAGE IN ISVD would allow much debugging.

Many important scripts, coded late in ISVD are not included in SUMMARY. On the whole, the script provides only suggestions for SUMMARY and an illustration of capability. Complete rethinking of its function in ISVD should precede any attempt to rescue the present form of the script.

ILLUSTRATION OF SUMMARY

SYSTEM: PLEASE TYPE YOUR I.D. NUMBER SO YOU CAN BEGIN.

INQUIRER: 4312

SYSTEM: NOW TYPE YOUR PASSWORD.

INQUIRER: (DOES SO)

SYSTEM: WELCOME TO ISVD, JOHN.
LAST TIME YOU WERE WORKING ON A DECISION ABOUT COLLEGES.
YOU LOOKED AT SCRIPTS ABOUT FACTORS IN COLLEGE ADMISSIONS,
COLLEGE PREFERENCES, COLLEGE DESCRIPTIONS, DATA.

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: IN THE COLLEGE PREFERENCE SCRIPT, THE CHOICES YOU MADE SUGGESTED
6 COLLEGES

UNIVERSITY OF MASSACHUSETTS
BOSTON UNIVERSITY
BOSTON COLLEGE
UNIVERSITY OF ARIZONA
UNIVERSITY OF MARYLAND
UNIVERSITY OF CALIFORNIA AT BERKELEY

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: YOU ASKED TO HAVE THESE COLLEGES DESCRIBED:

UNIVERSITY OF MASSACHUSETTS
UNIVERSITY OF MARYLAND
UNIVERSITY OF CALIFORNIA AT BERKELEY

YOU ASKED SPECIFIC QUESTIONS ABOUT

BOSTON COLLEGE

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: WHAT WOULD YOU LIKE TO DO?

ADDITIONAL Student Commands

It is not difficult to add student commands to GLURP, though it does require a reassembly of GLURP. Several commands seem particularly useful and are discussed below.

@HELP

@HELP could be used to help evolve the system if we asked the inquirer what help he was seeking when he used the command at unanticipated keyboards.

Instead of giving a standard "No hint provided" message, the system could branch to a script called HELP. The script would determine what help was expected and log that information along with step number and script name for author reference.

DEFINE

This script does not exist, but has been suggested numerous times by authors, inquirers, and field test supervisors. It would define a term for the inquirer provided the term was already in memory. If not, record of the term would be made for future insertion. The script could operate two ways: first, would be through the normal student command @DEFINE. Return would be to the CRT statement immediately preceding the KEYBOARD at which the command was used. The routine would operate cyclically for the inquirer so he could get several definitions with one use. The second mode would be under author control. The author would suggest the word for definition -- inquirers would not discriminate the text from their current script.

Probably the most effective way to provide definitions is as descriptors in a set of records in INQTABLE file. They could then be introduced or changed on line with no modification to the script body. Some mechanism for handling synonyms and different word endings would have to be employed. This might be a place to try a phonetic algorithm.

COMMENT

A comment script to elicit remarks about the scripts could serve as a powerful feedback device. I think that a natural language statement should be obtained to allow us the full range of English for expressing ideas. The statement could then be processed by a set of DECOMPS or by inquirer matching to fixed categories.

DECOMPS would allow us to categorize the comments according to emotional loading. We could probably also obtain a measure of goal

directedness, sense of agency, and possibly other important psychological measures. By concentrating effort on these DECOMPS, a compact way of measuring our main variables could emerge.

The matching technique, used before in COLCHANGEFACTORS, gives us the flexibility of human processing and a lot less work than the DECOMP method. Perhaps a good compromise would be to begin with a matching procedure (which could be in terms of a published list near the console) and use the natural responses to generate a set of DECOMPS later.

The comment script should have two entries, one for student command use @COMMENT, and another for author initiative.

The COMMENT script is intimately connected with MAIL in my mind.

MAIL

MAIL would provide a way for the inquirer to get feedback about his COMMENT's and problems he was having with the system. The "script" could be just as effective off-line as on. Actual letters could be handed to the inquirer before be logged or the MAIL could appear after he identifies himself.

The goal of MAIL is to tell the inquirer that we appreciate his COMMENT's and are using them to improve the system. I would assume that the type of MAIL would depend on the sophistication of the user. For some saying that "we missed a colon in a pop top" would just be confusing or amusing, though it might be the best MAIL for another. Since these MAIL messages will be written individually, they can be quite personal.

Another use of MAIL would be to provide new information about the system to inquirers who no longer get the introductory materials. Scripts recently debugged and connected to the system could be recommended to appropriate inquirers through MAIL.

#(SCR)FACT Vectors

These vectors have bits corresponding to each factor in our data bases. Typical of the factors are geography for college and salary for jobs. The bit positions are decoded using tables 16-19 of INQTABLE file. Those four tables are reproduced below.

DECODE TABLE - COLFACT - LIST 16

CODE	
Ø1	Geography
Ø2	Type of College (Public/Private-Religious Affiliation)
Ø3	Sex of Study Body (Men, Women, Coed)
Ø4	Size of Student Body
Ø5	Type of Programs (Liberal Arts, Preprofessional, etc.)
Ø6	Financial Aid Available
Ø7	Location (Urban, Rural)
Ø8	Special Courses of Study
Ø9	Extracurricular Activities
1Ø	Academic Standard for Admission
11	Cost
12	Housing

DECODE TABLE - OCCFACT - LIST 17

CODE	
Ø1	Physical Demands
Ø2	Working Conditions
Ø3	Hours
Ø4	Travel
Ø5	People
Ø6	Government Employment
Ø7	Education
Ø8	Salary
Ø9	Promotion
1Ø	Benefits
11	Job Stability
12	Discrimination
13	Future Need
14	Finding a Job
15	Requirements
16	Licensing
17	Abilities and Aptitudes

DECODE TABLE - TRAFACT - LIST 18

CODE	
Ø1	Geographic Location
Ø2	Type of School (Public, Private)
Ø3	Sex of Student Body
Ø4	Size of Student Body
Ø5	Program Offered
Ø6	Financial Aid
Ø7	Urban or Rural Setting
Ø8	Special Facilities
Ø9	Activities
1Ø	Admission Requirements
11	Cost
12	Housing
13	Graduation Requirements
14	Enrollment Date

DECODE TABLE - MILFACT - LIST 19

CODE	
1	Pay
2	Benefits
3	Education and Training
4	Location
5	Promotion
6	Special Units
7	Rank
8	Military Obligation
9	Reserve Programs

These vectors are the central focus of our major Monitoring routines and suggest many possible improvements. ANALYZE creates #(SCR)FACT vectors from a keyboard statement. EXPLORONE bases its interaction on #(SCR)FACT vectors. GOODORBADITEM creates and compares these vectors. Currently these are the only uses of #(SCR)FACT, but several suggestions are given below.

The factors correspond to factors used in the preference scripts. The choices made there could be encoded as #(SCR)FACT vectors. Templates offer choices based on the same factors. An inquirer's questions could be encoded. The DATA scripts essentially use the same factors. In each case VARSIMPLE#(SCR) recognizes the question. It could encode the choice of variables as well as list the sentences required.

The #(SCR)FACT vectors give us a powerful tool for tracking the dimensions an inquirer uses in considering schools (jobs). It also suggests a more powerful probe of his value system and his connection to reality. This involves an evaluation of his perception of a particular school (job) along each of the dimensions he mentions and its actual location in multi-dimensional space.

How could such an encoding be achieved? Most of the factors in #(SCR)FACT correspond to one or more level one vectors based on our data. In some cases the vectors have not yet been specified, but in others the preference script vectors are exactly what is needed. These level one vectors can serve as the reality measure. Let us consider some examples. Please refer to the COLFACT list 16 on page 56 and the Names of Level One vectors for colleges, which is reproduced below.

Factor one, geographic location, is related to the first six level one vectors.

1 New England colleges
2 Middle Atlantic colleges
3 Midwestern colleges
4 Southern colleges
5 Southwestern colleges
6 Western colleges

If an inquirer mentions geography in connection with a specific school we can test to find its regional location.

Factor two, type of college, is related to level one vectors 7-28. Most of the other factors show a similar correspondence. Those which already have level one vectors associated with them suggest definable vectors. As an example, consider cost, factor 11. No vectors relate, but the following variables obviously do:

tuition per year in dollars, resident
tuition per year in dollars, non resident
room and board per year in dollars, average
room and board per year in dollars, lowest
room and board per year in dollars, highest
other fees per year in dollars
total cost per year in dollars

These suggest categories for level one records such as:

TOTAL COST	$ Ø-1ØØØ
TOTAL COST	$1ØØ1-2ØØØ
TOTAL COST	$2ØØ1-3ØØØ
TOTAL COST	over $3ØØØ
TUITION	$ Ø- 5ØØ
TUITION	$ 5Ø1-1ØØØ
TUITION	$1ØØ1-15ØØ
TUITION	$15Ø1-2ØØØ
TUITION	over $2ØØØ

Careful statistical distributions across our data would provide more realistic categories, but these suggest the kind of correspondence we need.

We can look in greater depth at keyboard statements or even develop a small script to determine the inquirer's perception of a school's (job's) position on any dimension he mentions. The current ANALYZE script contains suggestions for these judgments, but is not at all complete:

For geography (step 2Ø1.ØØ) ANALYZE suggests near, far, and close as levels. While these are ambiguous, they can be mapped into New England location for near and any other location as far.

In GOBI we elicit a value judgment on the school as good or bad. That judgment presumably carries over to all the factors he mentions. We can therefore associate in storage the actual level of a school (job) on

a particular factor with the inquirer's positive or negative judgment. Once this association is made, further schools can be tested against his earlier judgments. If he says schools with high tuition are bad, he should like schools with low tuition. If he is inconsistent, we can Monitor that a draw his attention to it. Perhaps through such a scheme we can clarify his personal criteria for him. Even the process of focussing his attention on the levels of various factors will make him aware of his need to assimilate and collect data.

Other Types of Encoded Data

The utility of #(SCR)FACT vectors in the current version of ISVD arises because they provide a compact encoding of verbal behavior that is meaningful within the theory of ISVD. Clearly if other theoretically relevant constructs could be encoded from properly selected and prompted statements, they would prove fruitful in the context of ISVD.

Most higher level Monitoring functions currently in the ISVD relate to the #(SCR)FACT vectors. This fact reflects the deliberate choice made between two suggested alternatives. One possible course for Monitoring was to search out and provide ways to monitor as many different concepts of behavior as possible. The other course, which we chose, was to explore the potential of one concept as an illustration of Monitoring.

For more comment on Monitoring, see "Developing and Implementing Materials for Computer Assisted Instruction" by Richard Roman, ISVD Project Report No. 26.

ANALYZE

ANALYZE is a service routine that determines the factors mentioned by an inquirer in a statement contained in the KEYBOARD buffer. The statement is analyzed for factors related to #(SCR). Output from ANALYZE is a binary string called FVEC. The positions refer to the canonical names of the factors. FVEC is decoded by #(DC:INQTABLE:#(SCR)FACT:FVEC). FORMS NAMED KEY1 ... KEY#(N) are created for each factor recognized in the inquirer's

statement. KEY#(N) contains the words inquirer used to refer to factor #(N). The calling script must use and delete KEY1 ... KEY#(N) (use #(SD: KEY) and FVEC.

Output also includes LIST, a rule list of pairs associating a bit number with a root word of 4-7 characters. Structure is as follows:

,root1,number,root2,number2,.....,rootN,numberN

These lists are used only in EXPLORONE currently.

Action is via the LINK list. Forms named N, LIST, LIST1, LIST2, FVEC, KEY1 ... KEY#(N) will be deleted.

Note Extended notes precede the script.

Note The script logs each matched DECOMP #(KS:2) for easy reference. Valuable information on our effectiveness in natural language processing could be derived from this data.

Note ANALYZE runs for 1Ø-35 seconds with 1-3 users currently. Inquirers should be warned.

Note ANALYZE in no way modifies the Keyboard buffer contents.

ANALYZE currently recognizes only dimensions of concern. A more sophisticated and useful version would differentiate levels on the factors. Attention to user's statements would provide us aid in developing such a routine. One of the STINQ scripts will provide some of these data. (See SPECIFICATIONS for INQUIRER STATISTICS)

The current version of ANALYZE suffers from lack of ordering in the DECOMP rules. General words associated with a factor occur before the specific more meaningful words. "Get in" occurs before "tests" in the admission requirements DECOMP. This makes reporting the matches sound foolish and makes the routine seem ineffective.

The fact that the script writes the keyboard segments matched on the logtape allows us to collect a large amount of experience from our past field testing. The specifications for logtape processing include directions for obtaining this information in usable form.

The discussion of EXPLORONE contains suggestions for a script to help debug ANALYZE.

REALITY FEEDBACK

This routine has not been coded, but is suggested as an application of the ideas proposed under #(SCR)FACT. The purpose of the routine is to strengthen the inquirer's grasp of the real world and to engage him in a rapid feedback system.

One possible variable here would be tuition costs at college. The machine could randomly choose a college and present its tuition, asking the inquirer how he judged that value -- high or low. The true answer would then be presented. When inquirer was regularly correct, the categorization could be adjusted more finely high, low, or average. Eventually he could learn a five category system. With this new perspective he could look at his own choices in GOODORBADITEM.

Similar interactions in other areas could help him learn to categorize. Perhaps less dried evidence could also be used: you might give the description of what a man does in a job, and ask for a prediction of salary. This kind of interaction could develop a sense of the texture of the world of work or colleges.

Of course, you would want to measure the effectiveness of examples and sequences of instruction. Introducing appropriate variety and considering background variables as well as interaction history would allow such testing.

This routine would be an excellent one for an early session. So often we heard the complaint that inquirers didn't learn anything during the first few sessions. This routine would impress them with speed of data access.

Templates

Template scripts look simple, but mask considerable difficult coding. A template is designed to describe an item in the data base. It is put together as a set of sentence records which are also used [illegible] data script network.

Sentences usually deal with one topic and may contain considerable logic to deal with anticipated contingencies of the data base level zero records. Rules are of the form "If item is coded 9, skip words and 9." Problems arise when the actual coding does not correspond to the coding specifications.

Each data base has a sheet called DATA File, codes. These specify all legitimate codes for each data item. Unfortunately, some codes left blanks instead of writing zeros; some did not put in leading zeros in a wide field. The template sentences, as implemented assumed correct coding of data items. This causes blanks or sections of sentences to print confusing the inquirer.

Several courses of action are open: we could correct each sentence as it shows itself to make incorrect assumptions. This would be tedious and difficult since you can often be mistaken about the cause of an error. A second possibility is to recode and check each data item to be sure it corresponds to the allowable codes. Either the code sheet or the data could be changed; certain sentences could be rewritten in one session.

A third possibility is to use some existing computer programs to modify the data. TAPJNK can accomplish this task. Since no one knows how to use TAPJNK, I'll describe the procedure here. I'll assume you know how to create an HF deck for simple choices. Obtain the HF decks used to process the data. Check each card to see that it corresponds to the file specifications. If not, replace it by one that does. Run TAPJNK to obtain a list of error messages of the form: "_____ is unacceptable input for descriptor _____ in record _____." This list gives you all the errors in coding the data. Most will fall into groups by descriptor. For each descriptor with an error message, figure out what additional codes were used. Change the file description to correspond. If possible, create a F card that maps the erroneous code into its originally allowed equivalent:

Examples of the above would be:

a 2 column field, allowed ØØ-99. Unacceptable input ΔΔ. Map ΔΔ → ØØ.

a 3 column numerical field ØØØ-999. Unacceptable value Δ94. Map Δ## → Ø##.

Unacceptable blank field, map into zero length field to achieve no value.

Any unacceptable field that can not be mapped logically into a legitimate code will cause an error in the associated sentence. Correct sentence to account for the error. Change the sentence record document and the file document to reflect your changes.

Templates currently use only simple logic involving one or two variables. Better descriptions would result from a larger viewpoint. Certain industrial groups or professional groups should not have hourly wage reported. They should not have union membership or working conditions reported since they are irrelevant. Musicians are exposed to noise and vibration, but it seems strange to say it. Examining a set of twenty recomposed templates would allow such judgments to take place and recoding to occur.

Note Sentence names give a clue to their use: any name without a number following it can be used alone. Roman numerals simply several sentences dealing with the same topic. Arabic numbers indicate that only the highest number causes output. The lower numbers merely set up the data. They should never be used unless the full set is used.

Note Something is wrong with the execution buffer logic that causes GLURP to report a buffer overflow as a missing right parenthesis. A larger buffer would help. Finding the error would help some. Unless it is corrected, several sentences that generate long intermediate strings will cause this message. Ann Taylor could not find the error in several days of work.

It should be possible to build a template that would describe only those factors corresponding to a bit vector, for example one created by ANALYZE. This could operate in conjunction with PREFCALLTEMP to provide an automatic template for an inexperienced inquirer.

Another kind of template script could provide the same sentence or variable for all schools (jobs) on a binary list such as one created by a preference script. This would in effect give a rapid look at one important aspect of the description. Both the current full sentence and a briefer form could be employed. The kind of thing I have in mind follows:

Salaries for some of your jobs are

baker	$1.25
molder	$2.50

plater	$ 9.00
batter mixer	$ 8.75
blast-furnace keeper	$14.00

A series of slides could be controlled by such a template as well as giving vivid pictorial representation of the job or school.

One improvement in the templates and DATA idea would be to store a binary vector for each data item referenced. The vector would have bits on it if that factor was referenced. The cits would be as in ANALYZE, [#(SCR)FACT, tables 16-19 in INQTABLE, see pages 56-59]. These vectors could be used in EXPLORONE to see if the inquirer actually is pursuing all the factors he claims are important. In GOBI, we could test whether he actually has seen data on the factors he mentions in his reason the school (job) is good (bad) for him.

OCCUPATIONAL TEMPLATE

SYSTEM: YOU WILL BE ASKED FOR AN OCCUPATIONAL TITLE. YOU MUST USE THE NAME PRECISELY AS WE HAVE IT RECORDED. IF YOU DO NOT KNOW ANY JOBS YOU COULD TAKE THE OCCUPATIONAL PREFERENCE SCRIPT WHICH WILL LET YOU SELECT A JOB WELL SUITED TO YOU. WOULD YOU LIKE TO TRY THAT SCRIPT?

INQUIRER: NO, I'VE JUST TAKEN IT.

SYSTEM: YOU MAY NEED HELP GETTING THE NAMES CORRECTLY. WHEN YOU ARE ASKED FOR A JOB TITLE, YOU MAY TYPE A CORRECT JOB TITLE, YOU WILL SEE A BRIEF DESCRIPTION OF THE JOB. THEN YOU WILL BE ALLOWED TO ASK FOR FURTHER INFORMATION.

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: THE AVAILABLE TOPICS ARE

REQUIREMENTS FOR THE JOB
WAGES AND BENEFITS
PERSONAL QUALITIES NEEDED
WORKING CONDITIONS

SYSTEM: YOU MAY ASK FOR EACH IN TURN OR @STOP. FOR TECHNICAL REASONS THIS INFORMATION WILL BE COLD AND FORMAL. FOR THIS ISVD APOLOGIZES. AS TIME GOES ON, IMPROVEMENTS MAY BE POSSIBLE. MEANWHILE PLEASE REMEMBER THAT THIS INFORMATION IS INCOMPLETE. THERE IS MUCH MORE TO KNOW ABOUT A JOB THAN WHAT YOU SEE HERE. IF THE JOB INTERESTS YOU STRONGLY, SEE FURTHER INFORMATION...

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: WHAT JOB DO YOU WANT DESCRIBED? YOU MAY TYPE THE NAME OF THE JOB OR ITS NUMBER.

INQUIRER: INFORMAL WAITER

SYSTEM: AN INFORMAL WAITER IS A SERVICE WORKER. PRIMARILY, AN INFORMAL WAITER IS INVOLVED IN BUSINESS AND DOES CONCRETE AND ORGANIZED WORK. ON THE JOB WAITERS TAKE ORDERS AND SERVE FOOD AND BEVERAGES.

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: IF YOU WOULD LIKE TO LEARN MORE ABOUT THIS JOB, PLEASE INDICATE A TOPIC THAT INTERESTS YOU. LATER YOU CAN PICK ANOTHER. OTHERWISE, YOU MAY RETURN TO YOUR JOB LIST.

1. REQUIREMENTS FOR ENTRANCE AND ADVANCEMENT
2. WAGES AND BENEFITS
3. PERSONAL APTITUDES REQUIRED
4. WORK CONDITIONS AND OTHER DATA
5. TRY ANOTHER JOB
6. GO BACK TO WHAT I WAS DOING

INQUIRER: 1

SYSTEM: AN INFORMAL WAITER MUST AT LEAST COMPLETE JUNIOR HIGH SCHOOL AND SHOULD PREFERABLY GRADUATE FROM A VOCATIONAL HIGH SCHOOL. AFTER COMPLETING SCHOOLING AN INFORMAL WAITER USUALLY HOLDS ANOTHER JOB. PREVIOUS ACTIVITY MAY INCLUDE SPECIFIC TRAINING FOR THE JOB. UP TO 30 DAYS OF ON-THE-JOB TRAINING MAY BE INCLUDED. LOWER LEVEL JOBS LIKE BUSBOY MAY BE NECESSARY BEFORE BECOMING AN INFORMAL WAITER. HIGHER JOBS INCLUDE WAITER, FORMAL. NO INFORMATION AVAILABLE ABOUT ADVANCEMENT.

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: A HIGH SCHOOL STUDENT INTERESTED IN BECOMING AN INFORMAL WAITER SHOULD TRY TO TAKE COURSES IN ENGLISH, SALESMANSHIP, SOCIAL STUDIES.

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: IF YOU WOULD LIKE TO LEARN MORE ABOUT THIS JOB, PLEASE INDICATE A TOPIC THAT INTERESTS YOU.

INQUIRER: 4 [WORK CONDITIONS AND OTHER DATA]

SYSTEM: OF ALL THE PEOPLE IN THIS OCCUPATION 70-95% ARE WOMEN. AS AN INFORMAL WAITER YOU WOULD PROBABLY NEVER TRAVEL OVERNIGHT OR LONG DISTANCES. IF YOU WERE AN INFORMAL WAITER YOU WOULD OFTEN WORK WEEKENDS. YOU WOULD OFTEN WORK NIGHTS. AN INFORMAL WAITER IS SOMETIMES REQUIRED TO GET A LICENSE. THERE IS NO INFORMATION ABOUT UNION MEMBERSHIP. AS AN INFORMAL WAITER YOU WOULD BE SUBJECT TO NO HAZARDS. IF YOU WOULD LIKE TO LEARN MORE ABOUT THIS JOB, PLEASE INDICATE A TOPIC THAT INTERESTS YOU.

INQUIRER: 5 [I WANT TO TRY ANOTHER JOB.]

SYSTEM: WHAT JOB DO YOU WANT DESCRIBED? YOU MAY TYPE THE NAME OF THE JOB OR ITS NUMBER.

INQUIRER: BRIDAL CONSULTANT

EXPLORONE

This script is designed to deal with any data base. The particular data base is indicated as always by the value of SCR. When SCR is undefined, the script elicits and defines a value.

The central question of EXPLORONE is "What's important to you thinking about colleges (jobs)?" This question does not always work as expected. Inquirers do not know the appropriate response iniverse which is concrete, measurable factors that effect the college (job) choice. The HELP provided, as well as comments when an inquirer responds outside our legitimate universe,

assumes that he is trying to express feelings and has difficulty doing it. This mixed expectation further confuses inquirers who experience it.

Confusion of language is a consequence of the script changing as it developed from a general script about feelings to a strict encoding of factors we can measure. Severe rewriting of this text, including the initial question, is required to make the logical flow chart clear to the inquirers. The logtape will provide a list of responses made during the field test and help to clarify the necessary changes.

The idea of using ANALYZE on two different phrasings of a single idea has great power. We can strengthen our natural language processing by examining the logtape protocols produced here. We then have samples of inquirers struggling to make themselves understood. The specifications for logtape processing include a diagnosis of this routine.

The word "meanings" occurs to refer to the decoded list of recognized factors. This usage is never clear to the inquirer. Again this is a consequence of drastic revision of the script's conception as it was written.

One section of the script presents our understanding of his statement. If he disagrees on the interpretation of any phrase, he explains more fully. His new statement is ANALYZE'd again (668.Ø4). ANALYZE returns another factor vector. Sometimes the new vector contains the same bit as the original. This leads to statements like "Maybe when you say money you mean something like amount rather than cost, since you now say 'fees.'" Logic to guard against this would be desirable. Essentially it would involve using #(TB:FVEC:#(LISTNO):_____:) as well as the Boolean count (CB). (671.ØØ)

The paragraphs Ø75.ØØ - Ø82.ØØ are long and unresponsive for three keyboards. This kind of scripting is extremely detrimental especially in a script supposedly designed to elicit natural language response.

EXPLORONE functions in our system then in two ways. One is to elicit the actual factors an inquirer judges as important; the other is to test the ANALYZE script. These two functions are confused; the interaction seems hollow. We both presume to understand and are afraid to trust the routines. We then seem to be repeating the inquirer's statement to no real purpose.

I would suggest that both functions would be more effective if separated. One script would test ANALYZE with the intention of quickly and completely familiarizing the inquirer with the workings of that script. Such a script would provide us the debugging information and detailed information on the usage of words by our population. It would also assure us that the inquirer could make himself understood. Such a script would allow us to show inquirers the direct effect of their interaction to improve the system.

Periodic review of the logtape protocols of such a script would allow us to modify the routine. Messages in letter form or on-line would be available to thank inquirers and explain how the script was modified by their suggestions.

A sketch of the script protocol follows:

SYSTEM: WE ARE INTERESTED IN THE WORDS YOU USE TO DESCRIBE IMPORTANT FACTORS ABOUT COLLEGE. WHAT THINGS DO YOU CONSIDER WHEN JUDGING A COLLEGE?

INQUIRER: I WANT TO KNOW HOW MUCH IT WILL COST, HOW FAR FROM MY CITY THE SCHOOL IS AND IF I CAN GET IN WITH MY BOARD SCORES.

SYSTEM:

1	GEOGRAPHIC LOCATION	CITY
7	URBAN OR RURAL SETTING	FAR FROM MY CITY
10	ADMISSION REQUIREMENTS	GET IN
11	COSTS	COST

WE RECOGNIZED THE PHRASES ABOVE AND THEIR MEANINGS IN YOUR STATEMENT. DO ANY OF THESE IDENTIFICATIONS SEEM INCORRECT? TYPE IN THE NUMBER THAT IS INCORRECT AND WHAT YOU MEANT BY THE PHRASE. TYPE @HELP TO SEE YOUR STATEMENT AGAIN.

INQUIRER: 7 I MEANT WHERE IS IT.

SYSTEM: OK
"FAR FROM CITY" MEANS GEOGRAPHIC LOCATION, A SYNONYM FOR "WHERE IS". IS THAT BETTER?

INQUIRER: SURE

SYSTEM: 1 GEOGRAPHIC LOCATION CITY
10 ADMISSION REQUIREMENTS GET IN
11 COSTS COST

ANY OTHERS INCORRECT? TYPE @HELP TO SEE YOUR ORIGINAL STATEMENT.

INQUIRER: OK NOW.

SYSTEM: WE ALSO ARE CONCERNED THAT WE LEFT OUT SOMETHING IMPORTANT. PLEASE EXPLAIN MORE FULLY ANYTHING WE DID NOT RECOGNIZE. YOU SAID
I WANT TO KNOW HOW MUCH IT WILL COST, HOW FAR FROM MY CITY THE SCHOOL IS AND IF I CAN GET IN WITH MY BOARD SCORES.

INQUIRER: YOU DIDN'T MENTION BOARD SCORES. I WORRY ABOUT THE RESULTS OF THESE TESTS.

SYSTEM: 10 ADMISSION REQUIREMENTS SCORES
WE RECOGNIZED "GET IN" AS ADMISSION REQUIREMENTS BEFORE AND DID NOT MENTION "SCORES".

Suggestions for the coding of the script are given here. Statement S2 requires a decoding of the bit vector FVEC according to #(SCR)FACT. I would create a new decode table of the form:

Code	Value
1	1 geographic location -- #(KEY1)¬
2	2 type of college -- #(KEY2)¬
3	3 sex of student body -- #(KEY3)¬

etc.

This would give the desired printout without a lot of difficult, time consuming coding.

Statement S2 implies that the bit number is saved and tested in the second return. Appropriate alternate response would be "We still identify 'distance from city' as referring to urban or rural setting. We'll make a note of this error and try to correct it."

This kind of interaction frankly admits our problems and lets the inquirer see how the script works. He can then deal more effectively later

when he encounters inquiries about the factors he thinks are important.

The script to accomplish the second purpose can then be much simpler and effective. To elicit the factors an inquirer feels are important becomes more direct when he is familiar with the kind of statement we expect to handle. Such a script would avoid much of the conversation of EXPLORONE and assume that most identifications are correct. The possible ambiguities known from the other script, could be subject to direct questions. "When you say 'far from my city' are you referring to geographic location or the urban - rural setting of the school?" Most of these ambiguities can be eliminated by careful restructuring of ANALYZE. Those remaining could be frankly revealed.

EXPLORE SCRIPT

SYSTEM: WHAT DO YOU WANT TO DO?

INQUIRER: FIND A COLLEGE.

SYSTEM: WHAT'S IMPORTANT TO YOU IN THINKING ABOUT COLLEGES?

INQUIRER: I WANT TO KNOW HOW MUCH IT WILL COST, HOW FAR FROM MY CITY THE SCHOOL IS AND IF I CAN GET IN WITH MY BOARD SCORES.

SYSTEM: WE RECOGNIZED WORDS REFERRING TO URBAN OR RURAL SETTING, ADMISSION REQUIREMENTS, COSTS IN YOUR RESPONSE. IT'S ALMOST CERTAIN THAT THIS LIST DOESN'T INCLUDE ALL OF WHAT'S IMPORTANT TO YOU IN CHOOSING A COLLEGE. THE LIST MAY ALSO INCLUDE SOME MEANINGS THAT AREN'T IN YOUR STATEMENT AS YOU SEE IT.

INQUIRER: YES, IT DOES.

SYSTEM: IF YOU WANT TO KNOW HOW WE GOT ANY OF THESE MEANINGS OUT OF WHAT YOU SAID, TYPE IN THE MEANING YOU'D LIKE EXPLAINED, OR TYPE NO IF YOU DON'T WANT ANY MORE EXPLANATION. (URBAN OR RURAL SETTING, ADMISSION REQUIREMENTS, COSTS.)

INQUIRER: ADMISSION REQUIREMENTS.

SYSTEM: IT SEEMED YOU WERE TALKING ABOUT ADMISSION REQUIREMENTS WHEN YOU SAID "GET IN". DID YOU MEAN ADMISSION REQUIREMENTS?

INQUIRER: OF COURSE.

SYSTEM: ARE THERE ANY OTHERS YOU WANT EXPLAINED? URBAN OR RURAL SETTING, ADMISSION REQUIREMENTS, COSTS.

INQUIRER: URBAN OR RURAL SETTING.

SYSTEM: IT SEEMED YOU WERE TALKING ABOUT URBAN OR RURAL SETTING WHEN YOU SAID "FAR FROM MY CITY". DID YOU MEAN URBAN OR RURAL SETTING?

INQUIRER: NO.

SYSTEM: WHAT DID YOU MEAN WHEN YOU SAID "FAR FROM MY CITY"?

INQUIRER: I MEANT I DIDN'T WANT TO GO TOO FAR AWAY FROM HOME.

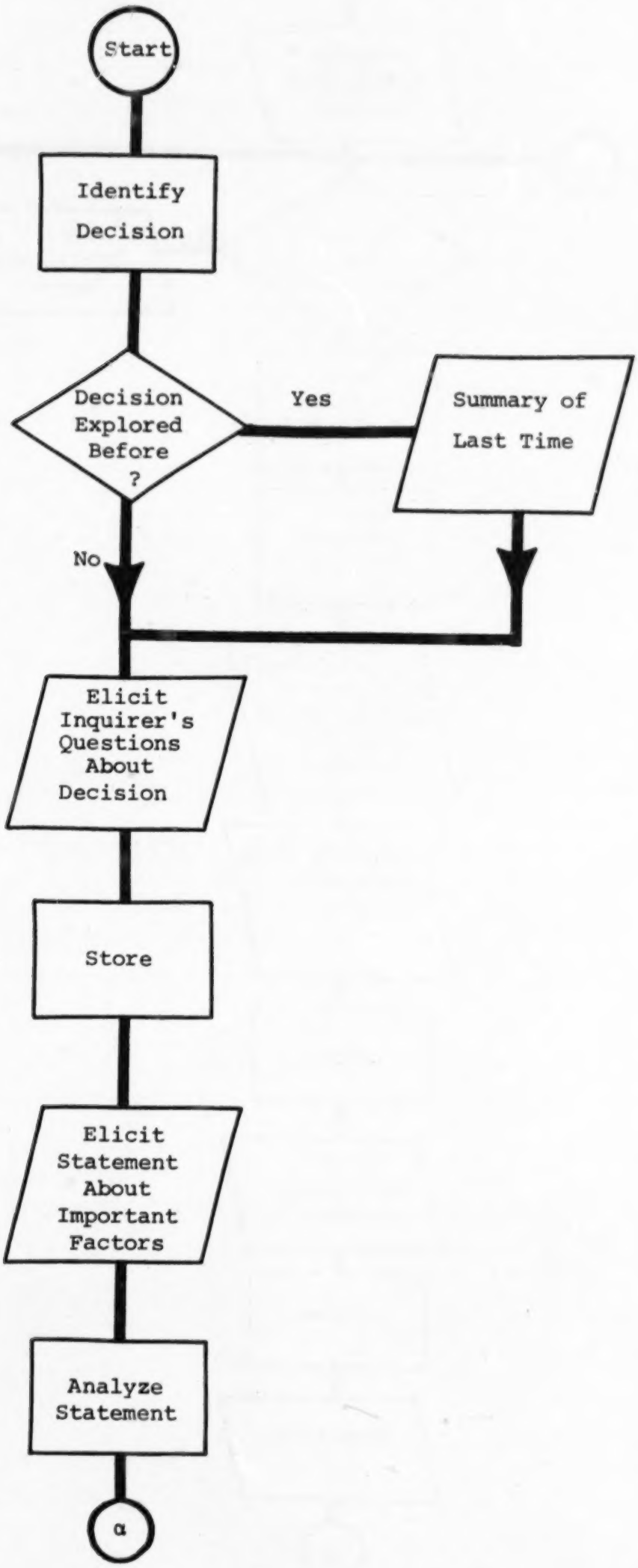
SYSTEM: MAYBE WHEN YOU SAID "FAR AWAY FROM MY CITY" YOU'RE TALKING ABOUT GEOGRAPHIC LOCATION INSTEAD OF URBAN OR RURAL SETTING, SINCE YOU NOW SAY "FAR AWAY FROM MY HOME".

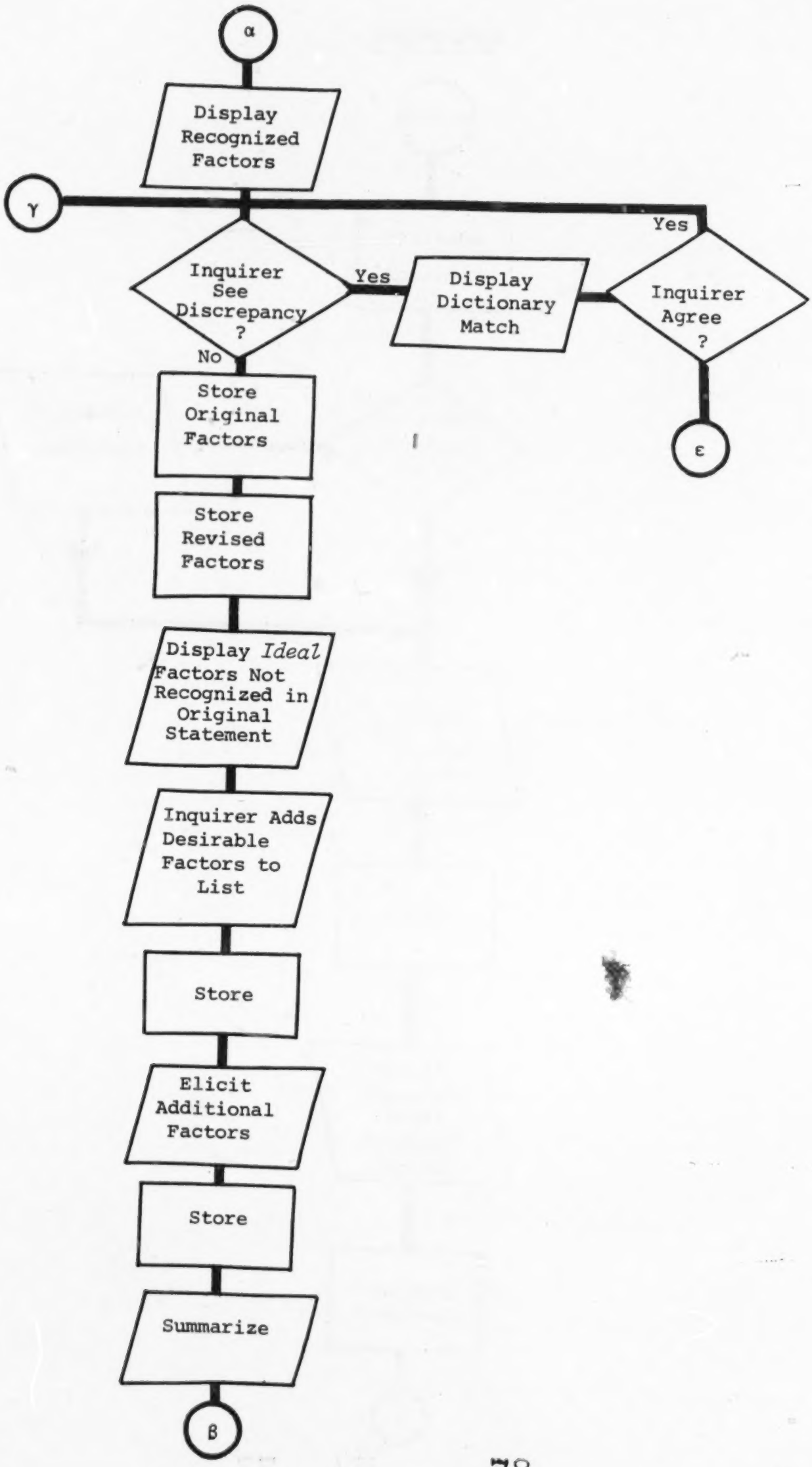
INQUIRER: CORRECT.

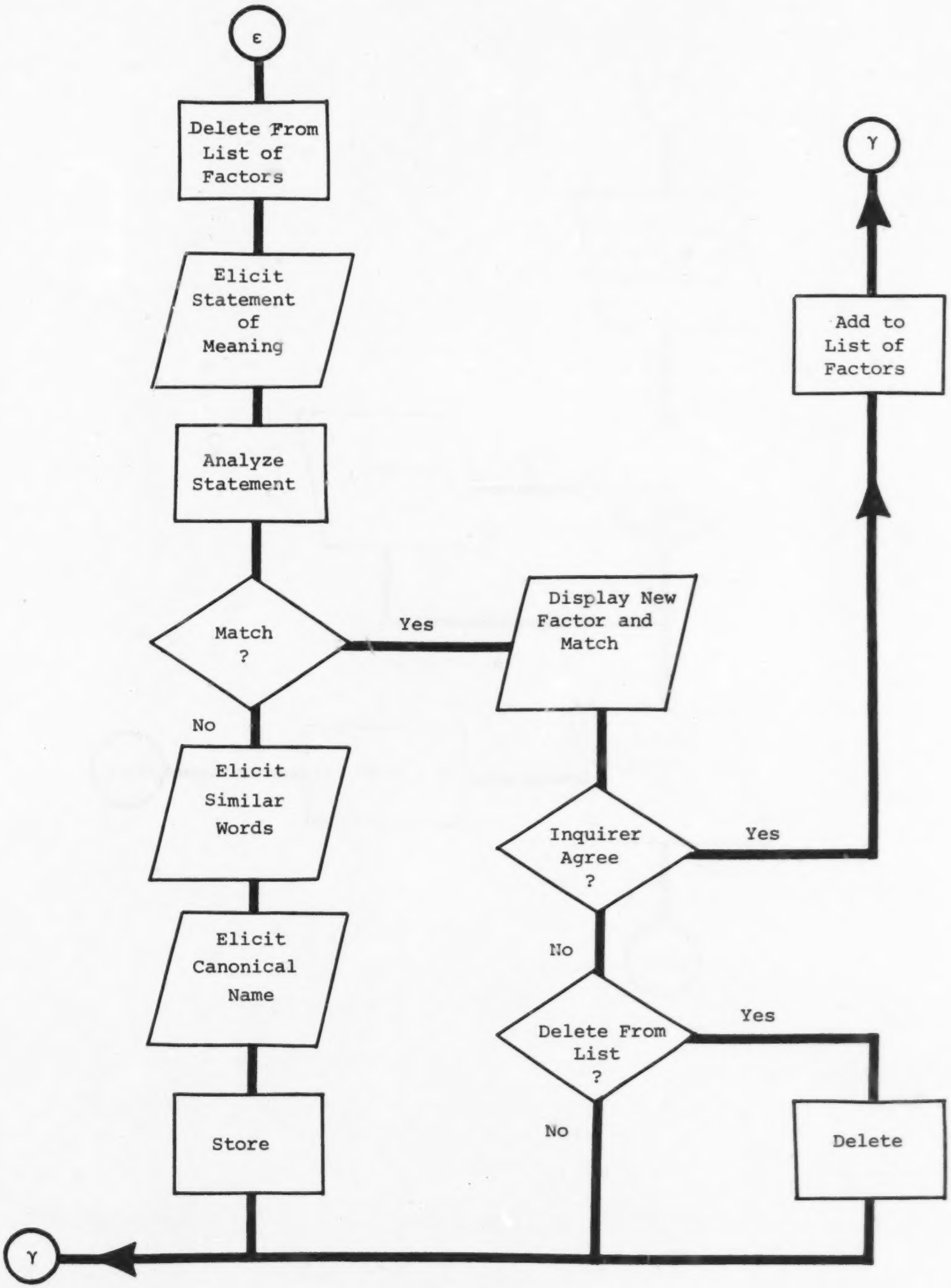
SYSTEM: GEOGRAPHIC LOCATION IS NOW ON YOUR LIST INSTEAD OF URBAN OR RURAL SETTING. ARE THERE ANY OTHERS YOU WANT EXPLAINED? GEOGRAPHIC LOCATION, ADMISSION REQUIREMENTS, COST?

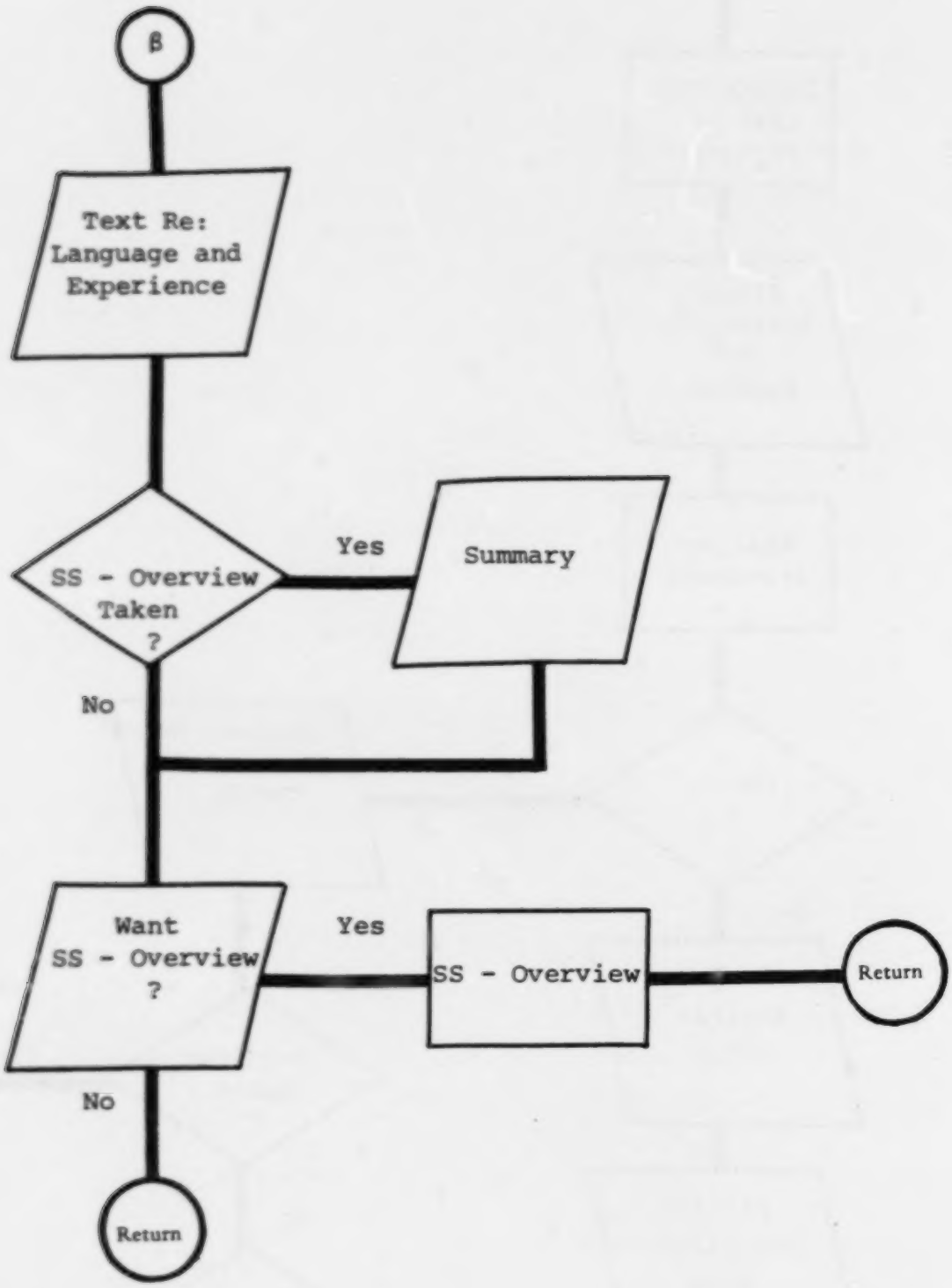
INQUIRER: NO. I WOULD LIKE TO GO TO THE DATA BASE TO GET SOME NAMES OF COLLEGES.

EXPLORONE


Start
Identify Decision
Decision Explored Before ?
Yes
Summary of Last Time
No
Elicit Inquirer's Questions About Decision
Store
Elicit Statement About Important Factors
Analyze Statement
α


α
Display Recognized Factors
γ
Yes
Inquirer See Discrepancy ?
Yes
Display Dictionary Match
Inquirer Agree ?
No
Store Original Factors
Store Revised Factors
Display *Ideal* Factors Not Recognized in Original Statement
Inquirer Adds Desirable Factors to List
Store
Elicit Additional Factors
Store
Summarize
ε
β


ε
Delete From List of Factors
Elicit Statement of Meaning
Analyze Statement
Match ?
Yes
Display New Factor and Match
No
Elicit Similar Words
Elicit Canonical Name
Store
Inquirer Agree ?
Yes
No
Delete From List ?
Yes
No
Delete
Add to List of Factors
γ
γ


β
Text Re:
Language and
Experience
SS - Overview
Taken
?
Yes
Summary
No
Want
SS - Overview
?
Yes
SS - Overview
Return
No
Return

GOOD OR BAD ITEM

GOBI tries to get an inquirer to consider his criteria for accepting and rejecting alternatives in concrete cases. It also incorporates the principles of review, selection among alternatives, and generalization from concrete cases. Attempts to Monitor at higher levels are contained here.

GOBI has been debugged and functioned in the field for several weeks. It is apparently one of our most successful routines in spite of occasionally confused wordings. The brevity of the main question in normal use increases its appeal. For most uses there are only two questions: "You just got data about #(CUTREC). Do you consider that school (job) good, bad, neither good nor bad?" "Why is #(CUTREC) a good (bad) school (job) for you?" The function of those two questions is to elicit data used later for comparisons. The job of stating reasons benefits the inquirer immediately.

When the inquirer has accumulated three good (bad) schools (jobs) in his data record he is asked to review the three and his reasons for selecting them. The three reasons are displayed; the inquirer generates a generalized statement of their commonality. This statement is ANALYZE'd and stored. When he has accumulated six schools (jobs) in either the good or bad record, he reviews his reasons and his previous generalization. Currently, these are the only times review occurs. Later implementation should continue to review whenever NMAX is divisible by three. This will create some coding problems where tests are made on the value of A, but they should not be too severe.

If the inquirer is working with a GOOD school (job), he is asked if any of the three (or six) stand out as the school (job) he will eventually choose. If one is his choice, he skips the review procedure and is shown his reason. Additional factors are then collected. The differences (items left out in the specific case) between his specific reason and his general reason given in EXPLORONE are displayed. If he has not taken EXPLORONE, he does so immediately.

In future versions, I would have the inquirer complete the review process before probing for a decision or a particular school. Both processes are important and should be used here.

The logic intended to display additional factors not recognized by the inquirer does not work since it is not reached by the program. Complementing a twenty four bit vector and decoding it will cause problems if the decode table does not go up to 24. High order bits should be turned off, the number of them depending on which data base is involved. See #(SCR)FACT, tables 16-19 in INQTABLE for more detail (pp. 56-59).

When seven items have already been put into a record, that record is full. Subsequent items must be substituted for previously recorded ones. This substitution is the reason each description of a school (job) is forced to exactly 14Ø characters. It also is the reason descriptor names are ANAL1, ANAL2,...,CRIT1,CRIT2,...,1,2,3,...,7 rather than using the school (job) number as part of the name. In this form each new descriptor can be exactly substituted in the space of its earlier value. The size of the record is as follows:

Descriptors	Name Count	Value Count	Size Each Descriptor	Total Size
1,2,3,...,7	1	4	1Ø	7Ø
ANAL1...ANAL7	5	4	14	98
CRIT1...CRIT2	5	14Ø	15Ø	1Ø5Ø
N	1	2	8	8
NMAX	1	2	8	8
			Record Size	1234

Adding always present 4 bytes, size is 1238. The size of each CRIT could be increased to 255 if that were desirable, however you would then have to be extremely careful of printing the description with no additional characters. The extra space was left so some text and three reasons would simultaneously fit on the screen. No one has ever used the entire 14Ø characters. New items are inserted in place of an item selected by an inquirer. The total number of items available at any one time never exceeds seven. This selection of what to discard is regarded as an important process, helping inquirers to realize that machines (and they) have only finite storage capacity, and must generalize to store more information in the limited space.

The items currently discarded when a GOBI record is filled are actually lost to the machine and inquirers. It would be possible to save some lesser amount of data about these choices in a second order file if anyone can think of a use for such data.

The general statement made to summarize a set of three or six other specific statements should be compared with them. The ANAL vectors are the appropriate encoding. Among the possible tests are:

Let ANAL1,ANAL2,ANAL3 be the vectors.

Let GEN be the general statement.

GEN should be a subset of the logical or of ANAL1, ANAL2, ANAL3. That is, each bit in GEN should be on in at least one of the ANAL vectors. A buffer criterion would force it to be on in at least on pair. Logically this could be done either by Boolean operations or bit testing.

The Boolean method is as follows:

```
#(DS:ANAL12:##(ANAL1))
#(DS:ANAL13:##(ANAL1))
#(DS:ANAL23:##(ANAL2))
#(BA:ANAL12:ANAL2)
#(BA:ANAL13:ANAL3)
#(BA:ANAL23:ANAL3)
#(DS:ANAL123:##(ANAL12))
#(BO:ANAL123:ANAL13)
#(BO:ANAL123:ANAL23)#(TO:ANAL123:24)#(TF:ANAL123:24)
#(BC:ANAL123)
#(BA:GEN:ANAL123)
```

GEN now contains bits that were not on in any pair of statements. This is destructive of GEN.

More compact, but longer in execution would be:

```
#(LOOP:N:24:(
    #(TB:GEN:#(N):(#(DD:CNTR)
    #(LOOP:R:3:(
        #(TB:ANAL#(R):#(N):(#(AN:CNTR:1)))))
        #(CP:1:#(CNTR):(#(TO:ANAL123:#(N)))))))))
#(DD:N:R:CNTR)
```

ANAL123 again contains bits only if the bit is on in two or more of ANAL1, ANAL2 or ANAL3.

Consistency in selecting reasons could be noted here. Comparing ANAL1...ANAL#(A) would give the consistency. This could be saved as a Monitor of his consistency. See also SCPT and CCF.

The GOBI script operating on colleges could call COLCHANGEFACTORS to ask for explanation of changes in criteria. More generally when #(SCR)CHANGEFACTORS are all implemented, it could be called as appropriate. See CCF write up. Several places are natural for this comparison: the first description offered in GOBI and the set of factors chosen in the COLLEGE PREFERENCE script could be compared. A schedule of comparisons to show changes in factors should be included in the script. Perhaps the best criterion is of the type: ask about changes when there are two or more differences between the last compared vector and the current vector. An illustration of this criterion follows:

1	10110100000
2	10110000000
3	10111000000
4	10111000000
5	10111000100
6	10011000100
7	10000110100

Compare 1 and 3, 3 and 6, and finally 6 and 7. Exclusive or (BE) provides the relevant bit string of differing factors which can then be counted.

Currently the script network calls GOBI only after a template is used. I would suggest several additional calls. After a person receives a preference script list, but before he has any data, one item could be used in GOBI. The objective would be to point out the need for data before a classification could be made. Particularly interesting would be a change in opinion following a use of data. Whenever DATA is used to obtain sentences or more than one variable for a school GOBI could be called. I would avoid using GOBI after each variable since the object of DATA is to provide rapid access and should not be interrupted by another script. Another possible way to achieve the same goal without interfering with the action of DATA would be to have DATACALLS store the number of times an item is referenced as well as the names of the items. A quick check of the appropriate record would then allow calls to GOBI.

SAMPLE CLARIFICATION INTERACTION

SYSTEM: YOU JUST GOT SOME DATA ABOUT BOSTON UNIVERSITY. DO YOU CONSIDER THE COLLEGE:
GOOD
BAD
NEITHER GOOD NOR BAD?

INQUIRER: GOOD

SYSTEM: WHY DO YOU CONSIDER BOSTON UNIVERSITY TO BE A GOOD COLLEGE FOR YOU?

INQUIRER: IT'S CLOSE TO HOME AND I THINK I CAN GET IN AND I HEAR THERE ARE GOOD FRATERNITIES THERE.

SYSTEM: YOU'VE SAID THESE 3 COLLEGES LOOK GOOD TO YOU
1 TUFTS UNIVERSITY
2 BOSTON COLLEGE
3 BOSTON UNIVERSITY
DOES ANY ONE OF THESE COLLEGES STAND OUT AS THE ONE YOU'LL GO TO?

INQUIRER: YES

SYSTEM: WHICH ONE? (TYPE THE NUMBER)

INQUIRER: 1

SYSTEM: YOU SAID TUFTS UNIVERSITY LOOKS GOOD BECAUSE "IT'S NOT FAR FROM HOME AND IT'S NOT TOO EXPENSIVE." IS THERE ANYTHING YOU'D LIKE TO ADD TO YOUR REASONS FOR LIKING TUFTS UNIVERSITY?

INQUIRER: IT HAS GOOD FRATERNITIES.

SYSTEM: WHEN YOU WERE TALKING ABOUT COLLEGES IN GENERAL, YOU SAID GEOGRAPHIC LOCATION, ADMISSION REQUIREMENTS, COST WERE IMPORTANT. IN DESCRIBING WHY YOU LIKE TUFTS UNIVERSITY, YOU SEEM TO BE TALKING ABOUT GEOGRAPHIC LOCATION, COST, SOCIAL ACTIVITIES. WHAT ABOUT ADMISSION REQUIREMENTS, SOCIAL ACTIVITIES?

INQUIRER: THEY'RE OKAY.

SYSTEM: HAVE YOU LOOKED INTO THE ADMISSION REQUIREMENTS, SOCIAL ACTIVITIES OF TUFTS UNIVERSITY?

INQUIRER: YES

SYSTEM: HOW DOES THAT MAKE YOU FEEL ABOUT TUFTS UNIVERSITY? IS IT STILL THE COLLEGE FOR YOU?

INQUIRER: YES

SYSTEM: YOU SEEM FAIRLY CERTAIN THAT TUFTS UNIVERSITY IS THE COLLEGE FOR YOU. WHAT DO YOU PLAN ON DOING ABOUT GETTING TO TUFTS UNIVERSITY?

INQUIRER: SEE MY COUNSELOR, GET AN APPLICATION FORM.

COLCHANGEFACTORS

This script embodies several ideas illustrated only once within the ISVD. Among these are allowing an inquirer to judge his own response, building up a record of inquirer's responses to enrich the repetoire of the system, recognizing uniqueness of response in a special record, reviewing changes over time, and collecting statistics on inquirer characteristics.

This script, to my knowledge, has been used only once. A protocol from the script is available along with other script protocols. Essentially the script is a service routine that elicits an inquirer's reasons for changing his mind about the importance of specific factors in choosing a college. If a factor is either added or dropped, the inquirer is asked to explain. After the free response, he is asked to compare his reason with those of other inquirers. If he matches, his response is recorded in his own records and in the public files. If he believes his response is different from all others, he is asked if we may show it to other inquirers, thus improving the set of selections. The fact that he has responded uniquely is recorded for later reference.

The script CCF is unconnected to the script network currently, but several links are desirable. In order to accomplish a link, an author must define several strings:

FIRST	This is a binary vector, each bit corresponding to a college factor as defined in list 16 of INQTABLE, COLFACT. Bits are on if the inquirer chose that factor in the first script under consideration.
SECOND	This binary vector is his choices in the second script.
FIRST NAME	This is an English phrase recalling the context in which the choices in FIRST were made. The phrase is used in several statements listed below.
SECOND NAME	English phrase recalling the second context. The uses of these names are as follows:
step Ø03.ØØ	One comes from #(FIRST NAME), the second comes from #(SECOND NAME)
step Ø28.ØØ	You listed exactly the same factors as important to you in #(FIRST NAME) and #(SECOND NAME)
step Ø57.ØØ	You felt in #(FIRST NAME) it was important, but later in #(SECOND NAME) you did not mention it.

Appropriate phrases might be:

your first exploration of college
your most recent use of the preference script selecting a college
your description of why Harvard is a good school

The scripts that use ANALYZE automatically prepare a vector appropriate for use in CCF. The preference script contains the list of factors, and choices could easily be coded as a binary vector.

I would suggest that after each script that creates an appropriate binary vector is executed, control be passed to a script named SHOULDWEDOCCF (Should we do college change factors). That script would then contain all the logic necessary to decide to call CCF or not, and to set up the strings as appropriate. Some parameter or different entry point would establish the calling script. Centralizing these decisions in one script will make the logic easy to change, and allow more efficient coding.

Obviously a similar script could be written in each other data base. I suggest using #(SCR)CHANGEFACTORS as the name for each data base.

CCF contains a call to CONSISTENTFACTORSCOL, a script not yet written. The call is executed when CCF has detected three sets of identical

choices. The three sets may not be identical to each other. This logic is easily changed if a more reasonable system is devised Ø31.ØØ. See STORAGE IN ISVD for details on how to access the relevant information in CONSISTENTFACTORSCOL.

Presumably the fact that inquirer makes responses he judges original and unique is of some guidance interest. That information is stored and available for reference in SUMMARY or CLARIFICATION routines.

Note Ø3Ø.ØØ where it says Ø it should say (#(DS:T:Ø))

Ø71.ØØ should print only if some exist. This test would involve a comparison of FKID and MAX and a test of FLG, messy, but worth it.

Note The documentation at the top of this script and the comments are very helpful. If I were starting over, all scripts would list form names and records referenced in the beginning. The purpose of any multi-line TRAC statement would be spelled out in comment cards.

COLLEGE CHANGE FACTORS

SYSTEM: YOU HAVE DEVELOPED SEVERAL LISTS OF IMPORTANT FACTORS IN COLLEGE ADMISSION. ONE COMES FROM YOUR LIST OF GENERAL CRITERIA, THE SECOND COMES FROM THE PREFERENCE SCRIPT. IN THIS SCRIPT YOU WILL BE COMPARING FACTORS THAT YOU HAVE SAID WERE IMPORTANT TO YOU AT DIFFERENT TIMES, AND ALSO LOOKING AT POSSIBLE REASONS FOR ANY CHANGES IN WHAT YOU FEEL IS IMPORTANT. WHEN YOU USED THIS SCRIPT LAST TIME, YOU COMPARED YOUR LIST OF REASONS HARVARD UNIVERSITY WAS A GOOD SCHOOL TO YOUR GENERAL CRITERIA FOR A COLLEGE.

INQUIRER: (PRESSES SEND BLOCK)

SYSTEM: ON THE FACTOR GEOGRAPHY, YOU FELT IN YOUR LIST OF THE GENERAL CRITERIA IT WAS IMPORTANT, BUT LATER IN THE PREFERENCE SCRIPT YOU DID NOT MENTION IT. PLEASE EXPLAIN BRIEFLY WHY.

INQUIRER: I FORGOT ABOUT IT LATER.

SYSTEM: THE GENERAL REASONS, APPLICABLE TO ANY OF THESE FACTORS, FOR REMOVING A FACTOR ARE:

1) SOMEONE SUGGESTED THIS FACTOR IS NOT IMPORTANT.
2) I DECIDED THAT IT IS NOT IMPORTANT TO ME IN CHOOSING A SCHOOL.
3) I WAS CURIOUS THE FIRST TIME AND WANTED TO SEE WHAT WOULD HAPPEN.
4) MY LIST OF SCHOOLS IS TOO SMALL.
5) I WANT TO CONSIDER SCHOOLS WHICH FIT ALL FACTORS I CONSIDER IMPORTANT EXCEPT THIS ONE.

INQUIRER: (MAKES A HARD COPY)

SYSTEM: YOU SELECTED GEOGRAPHY IN YOUR LIST OF GENERAL CRITERIA BUT NOT IN THE PREFERENCE SCRIPT.

6) MY PARENTS SAY I DO NOT NEED TO GO TO SCHOOL NEAR HOME.
7) I FORGOT IT THE SECOND TIME.

LOOK AT THIS AND YOUR GENERAL LIST. TYPE THE NUMBER OF THE REASON LIKE YOURS OR "NONE" IS YOURS IS DIFFERENT. YOU SAID "I FORGOT ABOUT IT LATER."

INQUIRER: 7

SYSTEM: ON THE FACTOR SEX OF STUDENT BODY (MEN, WOMEN, COED), YOU FELT IN THE PREFERENCE SCRIPT IT WAS IMPORTANT, BUT BEFORE IN YOUR LIST OF GENERAL CRITERIA YOU DID NOT MENTION IT. PLEASE EXPLAIN BRIEFLY WHY.

INQUIRER: I GOT MARRIED.

SYSTEM: SOME GENERAL REASONS FOR ADDING A FACTOR ARE:

1) I WANT A SMALLER LIST OF SCHOOLS.
2) SOMEONE SUGGESTED THAT IT MIGHT BE IMPORTANT.
3) I HAVE CHANGED MY MIND AND NOW THINK IT IS IMPORTANT.
4) I WANTED TO KNOW HOW THIS FACTOR WOULD CHANGE MY ALTERNATIVES.

INQUIRER: (MAKES A HARD COPY)

SYSTEM: YOU SELECTED SEX OF STUDENT BODY (MEN, WOMEN, COED) IN THE

SYSTEM: PREFERENCE SCRIPT BUT NOT IN YOUR LIST OF GENERAL CRITERIA.
5) THE SCHOOL MEETS MY REQUIREMENT SO IT IS LESS IMPORTANT.
LOOK AT THIS AND YOUR GENERAL LIST. TYPE THE NUMBER OF THE REASON LIKE YOURS OR "NONE" IF YOURS IS DIFFERENT.
YOU SAID "I GOT MARRIED"

INQUIRER: NONE

SYSTEM: GREAT. MAY WE USE YOUR REASON TO SHOW OTHER INQUIRERS?

INQUIRER: YES

SYSTEM: GOOD, NEW ANSWERS HELP US IMPROVE THE SYSTEM. WOULD YOU PLEASE TRY TO MAKE YOUR REASON SHORTER AND MORE GENERAL. IF YOU THINK IT IS OK NOW, JUST PRESS SEND BLOCK.
YOU SAID "I GOT MARRIED"

INQUIRER: (PRESSES SEND BLOCK)

SYSTEM: ON THE FACTOR SPECIAL COURSES OF STUDY, YOU FELT IN THE PREFERENCE SCRIPT IT WAS IMPORTANT, BUT BEFORE IN YOUR LIST OF GENERAL CRITERIA YOU DID NOT MENTION IT. PLEASE EXPLAIN BRIEFLY WHY.

INQUIRER: NO REASON

SYSTEM: HERE ARE SOME POSSIBLE REASONS THOUGHT OF BY STUDENTS AND OTHER INQUIRERS. YOU SELECTED SPECIAL COURSES OF STUDY IN THE PREFERENCE SCRIPT BUT NOT YOUR LIST OF GENERAL CRITERIA.
5) I FORGOT
LOOK AT THIS AND YOUR GENERAL LIST. TYPE THE NUMBER OF THE REASON LIKE YOURS OR "NONE" IF YOURS IS DIFFERENT.
YOU SAID "NO REASON"

INQUIRER: 5

SYSTEM: ON THE FACTOR EXTRACURRICULAR ACTIVITIES, YOU FELT IN YOUR LIST OF GENERAL CRITERIA IT WAS IMPORTANT, BUT LATER IN THE PREFERENCE SCRIPT YOU DID NOT MENTION IT. PLEASE EXPLAIN BRIEFLY WHY.

INQUIRER: MY LIST WAS TOO SMALL.

SYSTEM: YOU SELECTED EXTRACURRICULAR ACTIVITIES IN YOUR LIST OF GENERAL

SYSTEM: CRITERIA BUT NOT IN THE PREFERENCE SCRIPT. LOOK AT THIS AND YOUR GENERAL LIST. TYPE THE NUMBER OF THE REASON LIKE YOURS OR "NONE" IF YOURS IS DIFFERENT.
YOU SAID "MY LIST WAS TOO SMALL."

INQUIRER: 4

SYSTEM: THAT WAS THE LAST FACTOR YOU CHANGED. YOU ARE RETURNING TO THE PREVIOUS SCRIPT.

SELFCONCEPTSPROFILINGTECHNIQUE (SCPT)

This script is a computerized version of the second part of O'Mahoney's procedure. The pictures have already been examined and dimensions determined by the inquirer. The script follows the following logic: [see Chart 8, pp. 90-91.]

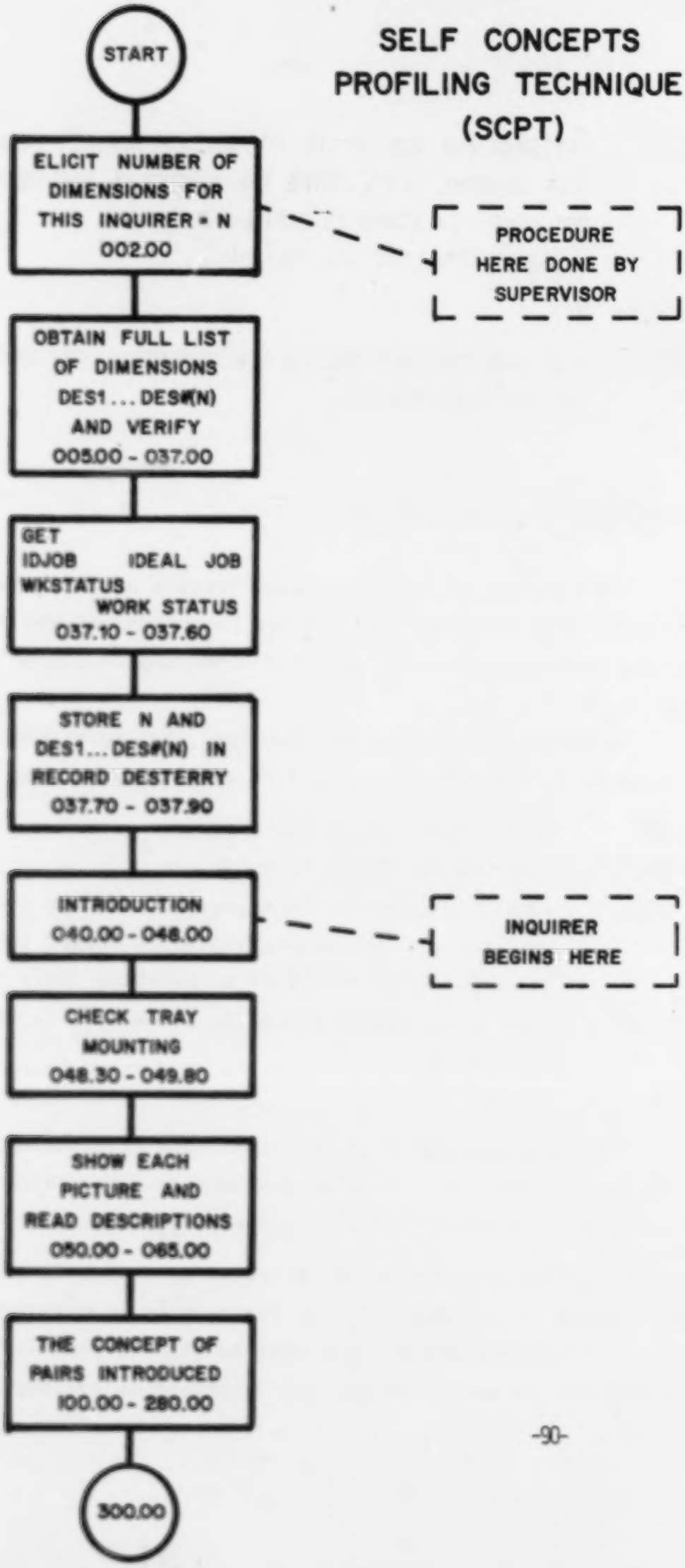
SCPT was implemented and debugged, but never used in the field test. All aspects have been tested and work with the following exceptions:

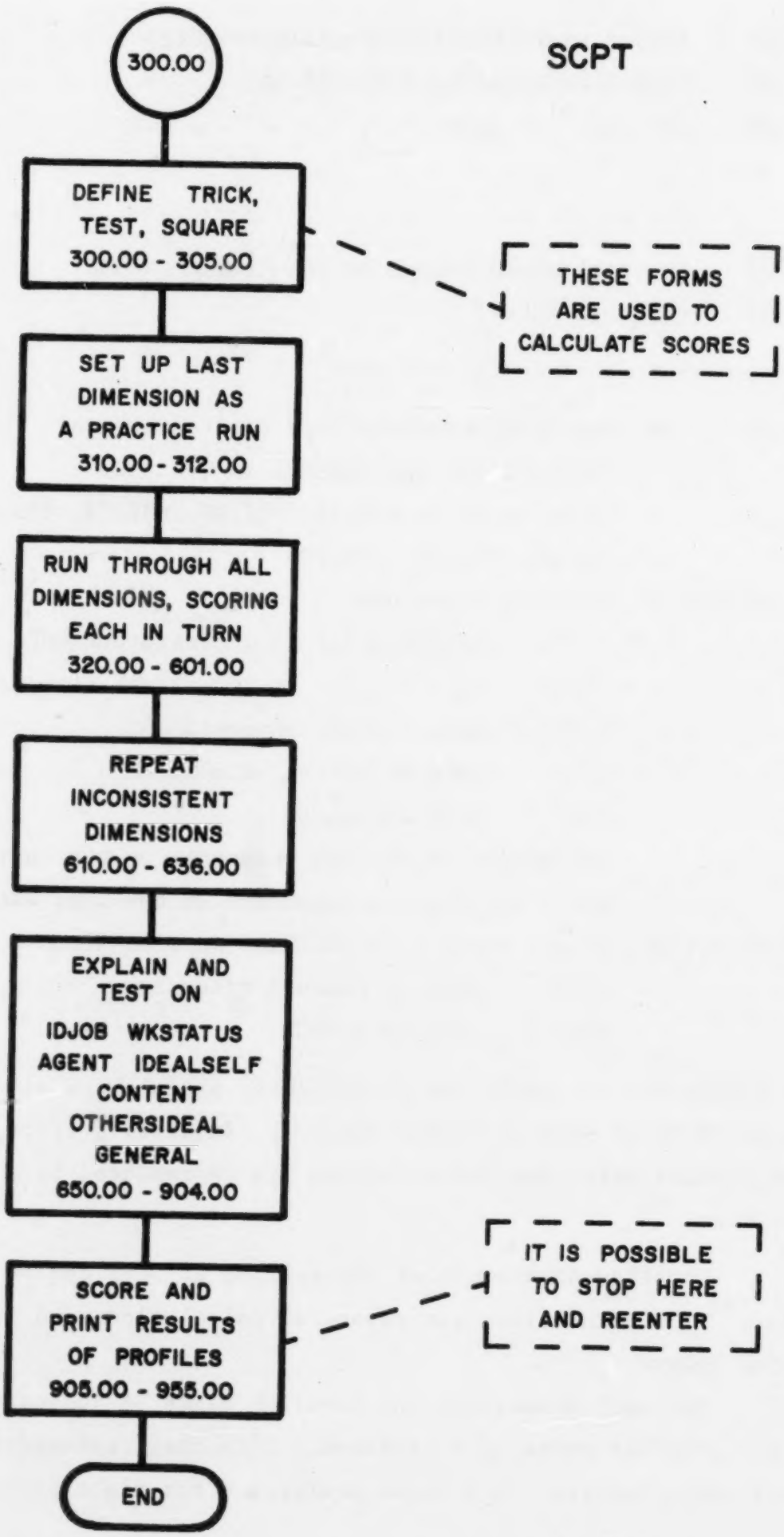
Ø48.ØØ	card B seems to be meaningless
Ø6Ø.ØØ	card 22Ø.ØØ should be moved
315-322	about 4Ø characters must be removed from text. Likely suggestions are decreasing blanks in 322.ØØ to 25 instead of 6Ø and shortening 317.ØØ,DE by deleting "this is the last slide." This is necessary since the screen overflows with the full text on it.
771.00	TRY not TYR.

The implementation of SCPT uses some programming devices unusual in the script network: internal subroutines, bit string manipulation and multiple entry points. Let's consider these in turn:

Internal subroutines are a device for using one piece of programming several times instead of copying the same script step in different places. The internal subroutine can be used with or without parameters that change the data the routine manipulates. The technique of calling used here is as follows:

SELF CONCEPTS PROFILING TECHNIQUE (SCPT)




SCPT
300.00
DEFINE TRICK, TEST, SQUARE 300.00 - 305.00
THESE FORMS ARE USED TO CALCULATE SCORES
SET UP LAST DIMENSION AS A PRACTICE RUN 310.00 - 312.00
RUN THROUGH ALL DIMENSIONS, SCORING EACH IN TURN 320.00 - 601.00
REPEAT INCONSISTENT DIMENSIONS 610.00 - 636.00
EXPLAIN AND TEST ON IDJOB WKSTATUS AGENT IDEALSELF CONTENT OTHERSIDEAL GENERAL 650.00 - 904.00
SCORE AND PRINT RESULTS OF PROFILES 905.00 - 955.00
IT IS POSSIBLE TO STOP HERE AND REENTER
END

```
100.00    #(DS:PARAM1:VALUE1)#(DS:PARAM2:VALUE2)
110.00    #(DS:BRANCH:120.00)#(GO:300.00)
120.00    CRT text continues
  .
  .
  .
300.00    internal manipulation on PARAM1 and PARAM2
315.00    #(GO:#(BRANCH))
```

Some internal routines in SCPT are:

315.00-318.00 Put slide commands on a cleared screen. There are no parameters here, only BRANCH is defined.
Calls occur at 310.10, 312.00, 630.00, 665.00, 686.00, 703.00, 728.00, 753.00, 778.00.

800.00-849.50 Parameters include:
STEP1 return point if inconsistent once
STEP2 return point if inconsistent twice (not used)
LAST name of last dimension
THIS name of current dimension
SEQ used as BRANCH
800.00-849.50 elicits judgments on the current dimension while scoring the judgments on the last dimension.

950.00-955.00 Prints results of profiles on dimension
THIS name of current dimension
SEQ used as BRANCH

Bit string manipulations are particularly useful here since the procedures are in terms of sets of binary choices. Inquirer prefers right or left in each picture pair, and those choices are represented by a bit string 36 bits long.

Detailed discussion of the scoring of SCPT can be found in O'Mahoney's thesis. The mathematics and method of calculation used in the script is presented below:

For each dimension, the inquirer makes 36 judgments of preference on all possible pairs of 9 pictures. O'Mahoney represents the choices in a triangular matrix: a 1 means preference for right picture of a pair. A

Ø means preference for left. The procedure is to count the number of zeros in a row and the number of ones in the column, sum them and square. Then sum the squares for a consistency number ranging up to 2Ø4.

To accomplish the counting, we use TRICK, a special bit string and complementing the CH string. Complement CH and call it Z, in our example Z =

ØlØØlØØlØllØØllØlØØlØlØØllØØlØØlØlØ

To count the number of zeros in the first row of CH is to count the number of ones in Z the complement. We must select the first 8 positions of Z and count the ones. The technique used in the script is to Boolean and Z with 11111111ØØØØØ... and then count the remaining ones.

right picture		1	2	3	4	5	6	7	8	9	1's in row	Ø's in row	sum	s
left picture	1	-	1	Ø	1	1	Ø	1	1	Ø	3	-	3	
	2		-	1	Ø	1	Ø	1	1	Ø	3	1	4	
	3			-	Ø	1	Ø	1	1	Ø	3	1	4	
	4				-	1	Ø	1	1	Ø	2	1	3	
	5					-	Ø	1	1	Ø	2	4	6	
	6						-	1	1	Ø	1	-	1	
	7							-	1	Ø	1	6	7	
	8								-	1	-	7	7	
	9									-	-	-	-	

Since it is difficult to represent triangular or multidimensional arrays in TRAC (GLURP), the script uses a linear bit string formed by packing the rows end to end; the array above is

1ØØlØllØlØlØØllØØlØllØlØllØØllØllØlØl

This is the inquirer's choice vectors. All choice vectors in the script are named CH..., e.g., CH1, CH2, CHWKSTATUS. For the first picture (row and column) this number is the correct total. For picture 2 we use

Z Boolean and ØØØØØØØØ11111111ØØØØØ...

CH Boolean and 1ØØØØØØØ...

Each picture is treated similarly.

TRICK is composed of a set of strings separated by asterisks that perform the necessary selection from CH and its complement. The coding

669.5Ø-671.ØØ performs the manipulation to make a string CNT that contains each picture total separated by spaces. Squaring is performed by segmenting on numerals Ø, 1...8 (TEST) and inserting their respective squares (SQUARE). The sum of squares is computed by successive use of PT (pop top) 815.ØØ-818.ØØ. The full value of TRICK, in hexadecimal is listed below for reference with asterisk as separators.

FF*ØØ*	picture 1
ØØFE*8Ø*	picture 2
ØØØ1F8*Ø48Ø*	picture 3
ØØØØØ7CØ*2Ø4Ø*	picture 4
ØØØØØØ3C*1Ø2Ø84*	picture 5
ØØØØØØØ38Ø*Ø81Ø422Ø*	picture 6
ØØØØØØØØ6Ø*Ø4Ø82112*	picture 7
ØØØØØØØØ1Ø*Ø2Ø41Ø894Ø*	picture 8
ØØ*Ø1Ø2Ø844BØ*	picture 9

The profile scoring consists of matching the choice vectors for each dimension against the choice vectors for various self images. The score for each dimension is the number of positions that match. Internally this is computed as the Boolean exclusive or BE.

Note A large number of complications arise in this program because the length of Boolean vectors is always a multiple of 8 bits in our implementation. Lengths are not determined by the will of the programmer, but must be forced by turning on an appropriate bit. Since all choice vectors should be 36 bits long we turn on bit 37 and then turn it off again to adjust the length to 4Ø. These extra 4 bits cause difficulty in counting, and must be subtracted from some totals (see 8Ø1.ØØ).

Multiple entry points in a script require that all information needed be stored on disc and fetched at the appropriate place. The SCPT does that at three places: one entry allows a supervisor to input the number of dimensions and their names. Another allows the inquirer to make his judgments on the dimensions. The last entry is used to report the scores from a session.

There are no explicit links to SCPT in the script network, although there are some potential links. One is suggested in the script IMINOR and expanded in the script TERRY.

While the SCPT script insists that the only meaning of a dimension is what the inquirer means by it, we can still recognize a certain commonality to the use of language among people. Different subjects repeat dimensions that look the same according to external standards: smart, intelligent, and brainy would all be the same for rough categorization. Granted this rough and ready approach to the subtlety of usage in SCPT, the same crude approach we use throughout the ISVD, we can map certain dimensions into the OCCFACT vectors. Concern for any of these could be used to tie into any of the procedures that use those vectors for control or comparison.

Life Career Game

For the sake of completeness the protocol of the Life Career Game is included here. Extensive comments on Gaming and its relationship to the ISVD are contained in the literature. The Game as implemented shows little of the promise anticipated and would need to be extensively recoded to insert Monitoring through relevant storage and processing.

LIFE CAREER GAME

SYSTEM: YOU HAVE CHOSEN TO PLAY THE ISVD CAREER GAME. THE CAREER GAME ALLOWS YOU TO MAKE DECISIONS FOR ANOTHER PERSON STARTING IN THE TENTH GRADE, AND TAKING HIM THROUGH TEN ROUNDS WHICH REPRESENT TEN YEARS OF REAL TIME.

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: IN ORDER TO PLAY THE GAME YOU WILL GET (1) A PROFILE, WHICH CONTAINS INFORMATION ABOUT THE PERSON WHOSE LIFE IS BEING PLANNED, (2) A GRADE TRANSSCRIPT, ALREADY CONTAINING A RECORD OF THE GRADES HE HAS MADE IN HIS FIRST TWO YEARS OF HIGH SCHOOL, AND (3) A WEEKLY SCHEDULE, WITH WHICH YOU WILL SHOW EXACTLY HOW YOUR PERSON SPENDS HIS TIME.

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: NOW LET'S PLAY THE GAME. YOU HAVE FIVE ROLES TO CHOOSE FROM.
[THE ROLES ARE DESCRIBED.]
WHICH ONE DO YOU WANT?

INQUIRER: JIM PURDY

[THE FOLLOWING IS A SHORTENED FORM OF THE PROFILE THE STUDENT SELECTED.]

SYSTEM: JIM PURDY: JIM IS 17, CAREFREE, AND INTERESTED IN SPORTS. ALTHOUGH HE DOES NOT DO WELL ACADEMICALLY, HE ENJOYS SCHOOL BECAUSE HE SEES HIS FRIENDS AND HE CAN PARTICIPATE IN INTRAMURAL SPORTS. JIM'S FATHER, HOWEVER, HAS SAID THAT IF JIM'S MARKS DO NOT IMPROVE, HE WILL HAVE TO LEAVE SCHOOL AND START EARNING A LIVING. DURING THIS ROUND OF THE GAME, JIM'S TIME WILL BE SPENT IN EIGHT KINDS OF ACTIVITY. SOME OF THEM ARE VERY IMPORTANT TO HIM, AND HE WILL WANT TO SPEND RELATIVELY MORE TIME DOING THESE THINGS. OTHER ACTIVITIES WILL BE LESS IMPORTANT. YOU NOW MUST DECIDE WHICH ARE THE FOUR MOST IMPORTANT ACTIVITIES FOR HIM. REMEMBER THAT THE IMPORTANCE OF AN ACTIVITY TO JIM IS DETERMINED BY MANY THINGS.

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: KEEP IN MIND, ALSO, THAT YOU ARE CHOOSING THESE ACTIVITIES FOR JIM -- SO DON'T FORGET WHAT THE PROFILE HAS TOLD YOU ABOUT HIM.
THE EIGHT ACTIVITIES ARE:

1. SCHOOLWORK
2. PART-TIME JOBS
3. HOUSEWORK
4. ATHLETICS
5. HOBBIES
6. CLUBS AND SCHOOL ACTIVITIES
7. SOCIAL TIME WITH FRIENDS
8. RELAXING AT HOME

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: NOW TYPE IN THE NUMBER OF THE FOUR ACTIVITIES YOU HAVE CHOSEN AS MOST IMPORTANT TO JIM. SEPARATE EACH NUMBER BY A SPACE AND DO NOT USE PUNCTUATION.

INQUIRER: 2 4 5 8

SYSTEM: NOTICE THAT YOU FAILED TO PLACE SOCIAL TIME WITH FRIENDS ON THE LIST OF IMPORTANT ACTIVITIES. ALONG WITH SPORTS, JIM'S SOCIAL LIFE MAKES SCHOOL IMPORTANT TO HIM. REMEMBER, THE OBJECT IS TO MAKE DECISIONS AS IF YOU WERE JIM. DO YOU WANT TO CHANGE SOCIAL TIME WITH FRIENDS INTO THE IMPORTANT CATEGORY?

INQUIRER: YES

SYSTEM: GOOD. NOW LOOK AT YOUR COPY OF THE ACTIVITIES AND DECIDE WHICH OF THE FOUR YOU SAID WERE IMPORTANT YOU WISH TO REPLACE WITH SOCIAL TIME WITH FRIENDS. TYPE IN THE NUMBER.

INQUIRER: 8

SYSTEM: YOUR REVISED SET OF ACTIVITIES IMPORTANT TO JIM SHOULD READ AS FOLLOWS:

2. PART-TIME JOBS
4. ATHLETICS
5. HOBBIES
7. SOCIAL TIME WITH FRIENDS

IF THIS LIST IS INCORRECT, TYPE NO AND YOU CAN START OVER AGAIN.

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: EACH PART OF THE LIFE CAREER GAME HAS AN ASSOCIATED SET OF RULES WHICH MUST BE FOLLOWED OR ELSE YOU WILL LOST POINTS ... SOON YOU WILL SEE THE RULES OF THE SCHOOL SECTION OF THE LIFE CAREER GAME ON THE SCREEN.

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: [PRESENTS THE SCHOOL RULES AND GIVES PLAYER AN OPPORTUNITY TO SCHEDULE COURSE AND STUDY TIME.]

INQUIRER: [SCHEDULES THE FOLLOWING COURSES AND STUDY HOURS FOR JIM.]

INQUIRER:

COURSE	COURSE HOURS	HOURS OF STUDY
ENGLISH	5	5
SOCIAL STUDIES	5	3
SCIENCE	5	4
INDUSTRIAL ARTS	5	
PHYSICAL EDUCATION	3	
	23	12

TOTAL SCHOOL HOURS: 35

SYSTEM: [PRESENTS THE WORK RULES AND GIVES THE PLAYER AN OPPORTUNITY TO SCHEDULE PART-TIME EMPLOYMENT.]

INQUIRER: [SCHEDULES 15 HOURS OF PART-TIME WORK, DOING ODD JOBS, FOR JIM.]

SYSTEM: THIS PART OF THE LIFE CAREER GAME DEALS WITH WHAT YOU DO IN YOUR FREE TIME.

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: YOU WILL SOON SEE A CHART LISTING THE POSSIBLE LEISURE ACTIVITIES JIM CAN ENGAGE IN.

LEISURE ACTIVITY	TIME COMMITMENT
ATHLETICS	12 HOURS A WEEK
HOBBIES	MINIMUM: 6 HOURS
CLUBS	MINIMUM: 3 HOURS
FUN WITH FRIENDS	ANY AMOUNT
RELAXING AT HOME	ANY AMOUNT

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: LET'S CONSIDER JIM'S LEISURE ACTIVITIES FIRST. NOW THEN, IS JIM GOING TO PARTICIPATE IN SCHOOL SPORTS?

INQUIRER: YES

SYSTEM: WHAT SPORT WILL JIM PLAY?

INQUIRER: BASEBALL

SYSTEM: HOW MANY HOURS WILL JIM SPEND ON THIS SPORT?

INQUIRER: 12 HOURS

SYSTEM: IS JIM GOING TO SPEND TIME ON A HOBBY?

INQUIRER: NO

SYSTEM: IS JIM GOING TO PARTICIPATE IN ANY CLUBS?

INQUIRER: NO

SYSTEM: NOW, HOW MANY HOURS WILL JIM SPEND WITH FRIENDS AND IN SOCIAL ACTIVITIES NOT COVERED ABOVE?

INQUIRER: 15 HOURS

SYSTEM: HOW MANY HOURS WILL JIM SPEND RELAXING AT HOME?

INQUIRER: 7 HOURS

SYSTEM: YOU HAVE FINISHED ROUND ONE OF THE GAME. AT THIS POINT, LOOK AT YOUR COPY OF THE WEEKLY SCHEDULE AND DECIDE WHETHER OR NOT YOU WISH TO CHANGE ANY PARTS OF IT. IF YOU DO, TYPE CHANGE. OTHERWISE, YOU WILL NOW SEE YOUR SCORE.

INQUIRER: [INQUIRER'S SCHEDULE LOOKS LIKE THIS:]

	MON.	TUES.	WED.	THUR.	FRI.	SAT.	SUN.
8:30	EN	EN	EN	EN	EN		STUDY-SC
9:30	SC	SC	SC	SC	SC		STUDY-SC
10:30	SS	SS	SS	SS	SS		
11:30	STUDY-EN	STUDY-EN	STUDY-EN	STUDY-EN	STUDY-EN		
12:30-1:00	LUNCH					JOB	RELAXING
1:00	IN	IN	IN	IN	IN		AT HOME
2:00	PE	STUDY-SC	PE	STUDY-SC	PE		
3:00	ATHLETICS						
4:00							
5:00		STUDY-SS	STUDY-SS	STUDY-SS			
6:00-7:00	DINNER						
7:00	JOB	FRIENDS	JOB	FRIENDS	FRIENDS	FRIENDS	FRIENDS
8:00	JOB	FRIENDS	JOB	FRIENDS	FRIENDS	FRIENDS	FRIENDS
9:00	JOB	FRIENDS	JOB	FRIENDS	FRIENDS	FRIENDS	FRIENDS
10:00	END OF DAY						

SYSTEM: THIS IS THE PART OF THE LIFE CAREER GAME WHERE YOU GET A SCORE FOR EACH ACTIVITY ... THE COMPUTER HAS DETERMINED THE GRADES THAT JIM WILL RECEIVE, ON THE BASIS OF STUDY HOURS ASSIGNED BY YOU AND GRADES RECEIVED LAST YEAR.

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: THE SCHOOL SCORE IS EQUAL TO THE NUMBER OF A'S X 4, PLUS THE NUMBER OF B'S X 3, PLUS THE NUMBER OF C'S X 2, PLUS THE NUMBER OF D'S.
AT THIS POINT, YOUR SCHOOL SCORE IS 7 POINTS.
JIM'S GRADE AVERAGE FOR THE FIRST ROUND IS BETWEEN D AND C.
ALTHOUGH JIM'S AVERAGE IS HIGHER THIS YEAR, IT IS NOT ENOUGH SO TO WARRANT A BONUS. WITH THOUGHTFUL SCHEDULING NEXT ROUND, YOU COULD GET A 10 POINT BONUS.

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: IF PART-TIME WORK WAS ONE OF THE FOUR PRIORITY ACTIVITIES, ANYTHING LESS THAN 10 HOURS PER WEEK COST A 5-POINT PENALTY. EXTRA POINTS WERE AWARDED IF JIM'S FAMILY HAS A LOW INCOME.

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: IN COMPUTING THE ATHLETICS SCORE, WE ASSUME THAT ANYONE FOR WHOM ATHLETICS IS HIGH PRIORITY WILL GO OUT FOR A TEAM, MAKING HIM SUBJECT TO THE 12-HOUR MINIMUM RULE. ANYTHING LESS COSTS AN 8-POINT PENALTY. OTHERWISE, YOUR SCORE IS EQUAL TO THE NUMBER OF HOURS SCHEDULED.
YOUR ATHLETICS SCORE IS 17 POINTS.

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: FOR THE REST OF THE CATEGORIES, YOUR SCORE IS THE NUMBER OF HOURS SCHEDULED. IF A CATEGORY IS HIGH PRIORITY, THERE ARE MINIMUM HOURS NECESSARY, OR ELSE A PENALTY IS IMPOSED.
THE MINIMUM HOURS ARE:

HOBBIES	4
CLUBS	6
SOCIAL	9
FAMILY	12

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: YOUR HOBBIES SCORE IS 0 POINTS.
YOUR CLUBS SCORE IS 0 POINTS.
YOUR SOCIAL SCORE IS 15 POINTS.
YOUR HOME AND FAMILY SCORE IS 7 POINTS.

INQUIRER: (PUSHES SEND BLOCK)

SYSTEM: YOUR TOTAL SCORE FOR ROUND ONE, WITH JIM, IS 76 POINTS.
YOU DID A GOOD JOB OF PLANNING THE YEAR FOR JIM.
WE HOPE YOU HAVE ENJOYED PLAYING THE I.S.V.D. CAREER GAME.
WHEN MORE ROUNDS ARE WRITTEN, YOU ARE INVITED TO CONTINUE PLAYING.

EDIT - SEDIT

The EDIT-SEDIT pair of scripts is used to make on-line changes in scripts and data bases. Neither script can function alone because we are treating a script as a datum in the editing process.

HASM contains two script FPB's, one for SCRIPTSA, one for SCRIPTSB. One or the other is always used as the execution FPB, and any manipulation of it will cause errors of script flow. However, the FPB not current for execution can be manipulated through FETCH and STORE and REPLACE commands at will without messing up execution. EDIT and SEDIT operate from different FPB's and therefore we can manipulate any script as long as we choose the correct script to work from. EDIT itself actually makes the decision and transfers control properly when necessary. In using this facility, always start by @XEQ EDIT, not SEDIT.

Several strange things happen in EDIT. Under the most current operating system (07/22/69), TRAC removes all periods, question marks and exclamations from input. This makes it difficult to specify step numbers either for segmenting or location. The strange code 230.00-231.00 inserts the period again. Among their other uses, the string F and R allow input of periods and questions. They also allow formatting characters and edit change characters to be

inserted in strings. The general technique to accomplish this is to type in a symbol for the code rather than the code itself. Codes are as follows:

/T/	type
/FT/	format type
/Q/	?
/P/	. (period)
/E/	!
/FD/	format delete
/FI/	format insert
/I/	insert
/D/	delete
/MC/	move cursor
/RC/	reset cursor
/CL/	clear
/C/	, (comma)

Others that could be added are:

/NL/	null
/CR/	carriage return
/VT/	vertical tab
/HT/	horizontal tab
/SB/	start blink
/CB/	clear blink

An example of this feature in action follows:

```
SYSTEM:    #(DS:S:14Ø.ØØ)#(DS:R:14Ø.1Ø)#(PS:OK!)
           TYPE IN SEGMENT PLACES SEPARATED BY COLONS -- UP TO 6.
           USE TRAC FOR LITERAL : ( ).
           USE /P/ FOR PERIOD.

INQUIRER:  Ø/P/ØØ:/E/

SYSTEM:    #(DS:S:14*1*)#(DS:R:14Ø.1Ø)#(PS:OK*2*)
           YOU MAY NOW TYPE YOUR SUBSTITUTIONS.  IF YOU WANT TO SEGMENT
           AGAIN, USE @HELP. SEPARATE BY COLONS AGAIN.
           USE TRAC FOR LITERAL : ( ).
           USE /P/ FOR PERIOD.
```

```
INQUIRER:  1/P/ØØ:/Q/

SYSTEM:    #(DS:S:141.ØØ)#(DS:R:14Ø.1Ø)#(PS:OK?)
           IS THAT CORRECT
           A    YES, STORE IT
           B    NO, SUBSTITUTIONS AGAIN
           C    NO, SEGMENTS AGAIN
           D    FORGET IT, NEW STEP
           E    FORGET IT, NEW SCRIPT
           F    RE-EDIT BEFORE STORING
```

A number of improvements seem possible now: the script should have a short mode for experienced users. This would offer only brief hint of appropriate behavior rather than full requests. Some way of elicit the full request.

The use of TRACΔ#(for unbalanced parentheses should be clearer. The benefit is that you can handle unbalanced parentheses or parts of large expressions. The cost is that only one segment can be treated at a time.

Through experience it is clear that the same script, different segment is often searched or changed. It would help if EDIT remembered or extracted the root name and could take a new number as a suffix.

A facility to put in an arbitrary bit string would occasionally be helpful. Code which will convert from hexadecimal input to bit string follows:

```
D;1ØØ.ØØ;#(DS:TTAB:FOOOOEOOOFDOOFOCOFFBOFOOAOFOF
                   9OFFO8OFFF7FOOO6FOOF5FOFO4FOFF
                   3FFFOO2FFOF1FFFOØFFFF)
D;11Ø.ØØ;CRT TYPE IN A HEXDECIMAL STRING
D;12Ø.ØØ;KEYBOARD
D;13Ø.ØØ;#(DS:S:##(CK))
D;14Ø.ØØ;#(DS:CNT:Ø)#(DD:T)
D;15Ø.ØØ;#(CR:TTAB)##(IN:TTAB:##(CC:5:*9ØØ.ØØ):(#(PS:BAD)))
        #(LOOP:N:4:(
             #(T#(CC:TTAB:(#(PS:ERROR))):T:#(AN:CNT:1))))
        #(GO:15Ø.ØØ)
D;9ØØ.ØØ;#(DD:S)#(PS:T NOW HAS YOUR STRING)
```

The coding in step Ø58.ØØ determines if the final letter of the script name begins with a letter before M or M on. Those before M are in

scripts A, those M on are in scripts B. The testing is dependent on EBCIDIC letters.

Great care should be taken in modifying the EDIT scripts to distinguish active from neutral functions ##(#(. Any string typed in by an inquirer may contain embedded track functions.

Because of the structure of protective parentheses, it is possible to segment on colons by protection. As an example

(AB:C):D:(:) has 3 items in it and will create 3 classes of segments.

AB:C:D:(:) has 4 items. Using TRACΔAB:C:D:(:) you have only one segment type.

The data base editing section has not been well debugged and hardly even used.

At 285.ØØ the R$ and A$ DECOMP branches are quite general, and have caused one error. The text that responds to R$ is different than that for S$ and probably should alert the user, but it does not.

Steps can be edited as long as the length remains constant or is decreased. If the length is odd, it can be increased by one character. Attempts to add more than one character, even if apparently successful will cause steps not to be found. Most of the time an error message results. Under some circumstances it is possible to add large amounts of text or new step numbers. This can be done usually in the last script segment. The technique is as follows:

At the place you want to insert the text, put a branch to a step between the second to last and last stage of the script. In the example below, step 991.ØØ would be appropriate:

```
D;99Ø.ØØ;#(CLEANUP)
D;995.ØØ;#(XQ:#(PT:LINK):#(PT:LINK))
```

Now create 991.ØØ and other steps with numbers below 995.ØØ to complete the task and branch back. Then delete 995.ØØ and retype it. This is possible only if there are enough characters left in the last record for your steps. Space required is 12 bytes and length of step. See Use of Storage in ISVD for further details on the storage requirements.

Improvements to EDIT range from simple to complex. Allowing arbitrary length changes to steps maintaining the logical ordering within the script requires changes to HASM or at least to GLURP. Ideally HASM would automatically adjust the relative address pointers to preserve ordering. In cases where increasing length forced overflow of the data record, the record could be consolidated and retested. If still too long, a new record and some sequencing that allows inserting new continuation records between existing ones would be required. This is a difficult problem with the system designed as it is; on-line editing in its full sense is not really provided.

Another improvement would be to allow HASM to provide the searching for a step in various records through a subpart of the XQ logic (HASM function 65). This would use the table of contents to locate the step required by step 26Ø.ØØ.

Alternately the script could be changed to take advantage of the table of contents itself. The table of contents descriptor of a script must be present in every root and continuation record. It is called $$ and can also be addressed as 1+. Internally, the value of the table of contents has structure as follows:

bytes 1-2	w = hexadecimal count of width of each entry
bytes 3-(2+w)	first w characters of lowest step in root record
bytes (3+w)-(2+2w)	first w characters of highest step in root record
pairs of w character sets	first w characters of lowest and highest steps in successive continuation records

The end of the descriptor is signified either by running out of characters or a set of w characters with value WX'FF' code that will utilize the table of contents that follows. It should occur near 26Ø.ØØ and will be numbered accordingly. Assume

#(TYPE)	is A or B for scripts A or B
#(SCRIPT)	is root name of a script
#(STEP)	is step number to be located.

[see pp. 106-107 for Code and Flow Chart]

If this logic were to be implemented, you should allow user to specify either a root record or a specific continuation, saving the root

```
   D;26Ø.ØØ;#(FO:SCRIPTS#(TYPE):#(SCRIPT):$$:TOFC:(#(PS:NO TABLE! AMAZING)
A            #(GO:Ø5Ø.ØØ)))
   D;26Ø.Ø1;#(DS:SEG:ØØ)#(BO:TOFC:SEG)#(DD:SEG)

*            STEP 26Ø.Ø1 INSERTS THE SIGN FIELD IN THE HEXADECIMAL FIELD.
*            Ø HAS HEX VALUE X'FØ'
*            NOTE THIS DOES NO HARM EVEN IF SIGNS ARE ALREADY THERE.

   D;26Ø.Ø2;#(DS:W:#(CN:TOFC:2:*Ø5Ø.ØØ))
            #(DS:STP:##(STEP))#(SS:STP:.)#(DS:STP:#(STP:))

*            STEP 26Ø.Ø2 REMOVES THE DECIMAL POINT FROM STEP MAKING STP

   D;26Ø.Ø3;#(CP:3:#(W):(#(SS:TOFC:.)#(DS:TOFC:##(TOFC:))#(SN:W:1)#(CN:TOFC:2)))
   D;26Ø.Ø4;#(DS:STP:#(CN:STP:#(W)))

*            NOW ALL PERIODS ARE REMOVED AND W IS ADJUSTED
*            THE  CP FUNCTION DOES NOT WORK ON NON-NUMERICS

   D;26Ø.2Ø;#(DS:OLD:#(CN:TOFC:#(W):(#(PS:NOT THERE)#(GO:21Ø.ØØ)))
A            #(EQ:#(CC:OLD: :(#(PS:NOT THERE)#(GO:21Ø.ØØ)))
B            #(CP:#(STP):#(OLD):26Ø.4Ø)
C            #(CP:#(CN:TOFC:#(W):(#(PS:ERROR)#(GO:21Ø.ØØ))):
D                 #(STP):*26Ø.45)
   D;26Ø.25;#(FO:SCRIPTS#(TYPE):#(SCRIPT)#(CN:SEG:2:Ø#(CC:SEG)):#(STEP):OLD:
A            *26Ø.45)#(DS:SCRIPT:#(SCRIPT)#(CN:SEG:2:Ø#(CC:SEG)))#(GO:261.ØØ)
   D;26Ø.4Ø;#(CN:TOFC:#(W))
   D;26Ø.45;#(AN:SEG:1)
```

CODING SHEET

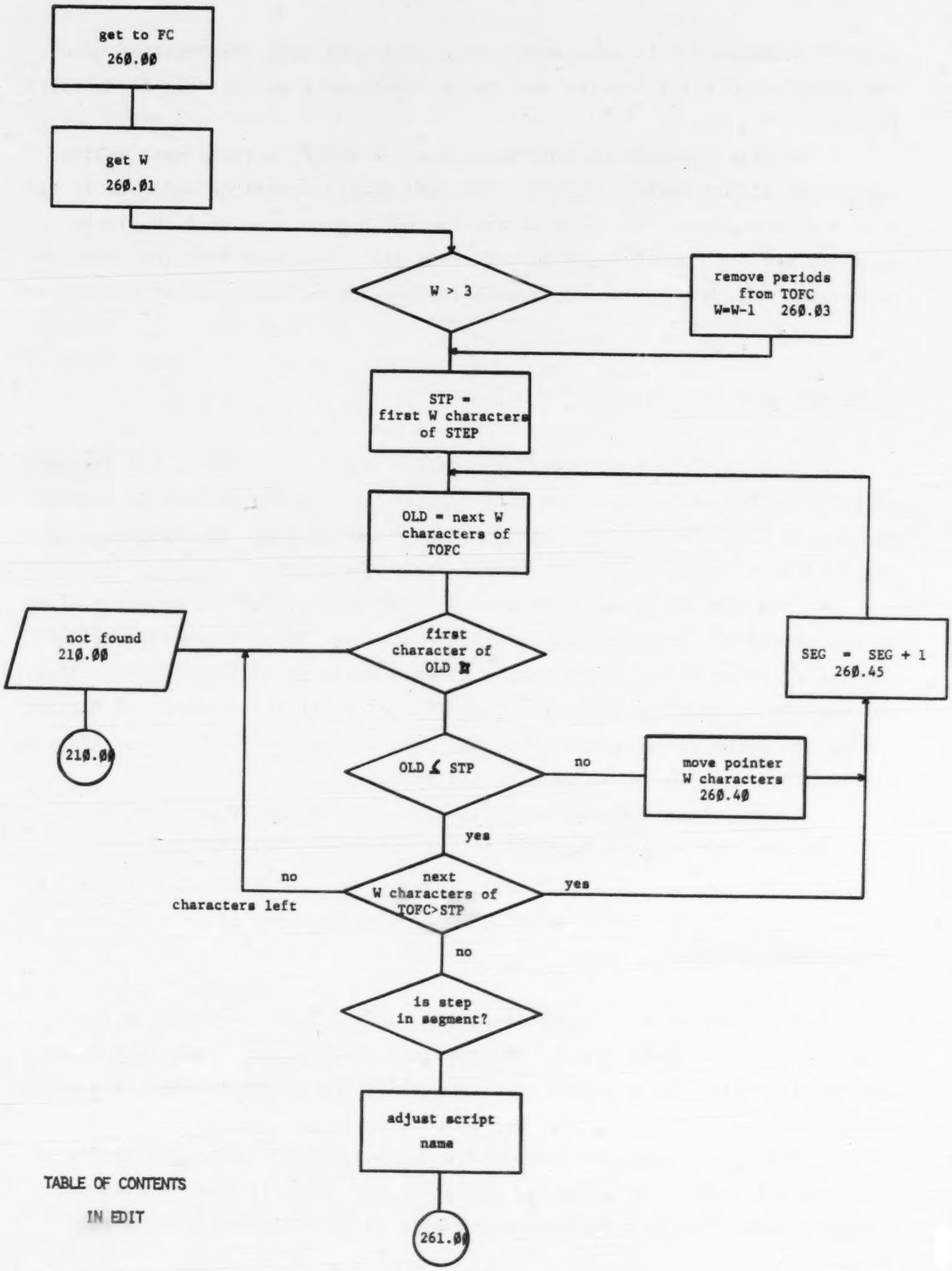


TABLE OF CONTENTS

IN EDIT

name (by calculation if necessary), as well as the last continuation used. The ideal would allow a naive user to believe that there is only one script-record.

To make possible for guidance staff to modify scripts more easily, especially simple textual changes, the EDIT script should be modified in the ways suggested above. A count of the current length of a step should be computed with the COUNT function and saved for comparison with the COUNT of new step. This would save accidental destruction of the order of steps.

SHOMP AND HOMP

These scripts have never been used. Their object is to set the home priority of various scripts and data records so they can be used in sorting records in PURGE format. The scripts keep track of their own progress and theoretically can be continued wherever they leave off.

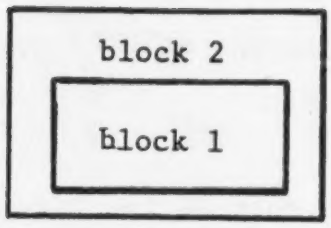
Unfortunately, this is not true. The BTSS system can crash or stop in the middle of a logical DASM store, causing the index to be updated and the record to be lost. Additional KEYBOARDS would provide stopping points, but require an operator on hand. Some thought about the tradeoff of storing counts for each record versus reporting some storage because of less frequent updates is desirable before running the scripts.

Consultation with the specifications for SH and RH set hom and read home priority is required before modifying this script.

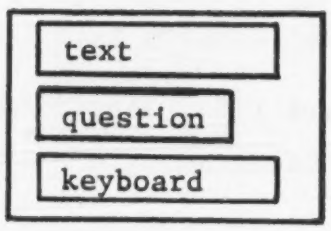
JOBCORPSREADINGTEST

This program is a multiple choice reading test resulting in a placement according to grade level. The material is excellent, well tested and accurately scaled for off-line use. No validation of the on-line version has taken place.

The script uses the SANDERS 720 capabilities of arranging blocks of text and selectively deleting and replacing it. This is done through the string LOCATE, 019.00. The screen normally is in 2 blocks, keyboard and text.



In JCRT, we create a third block. Block one is still the keyboard, block 2 is the text and block 3 contains one multiple choice question. Block 3 is changed until all questions referring to the text are answered; then block 2 is changed.



As an example, the text in Ø2Ø.ØØ-Ø21.ØØ looks like this:

```
 WHAT A DIFFERENCE THERE IS BETWEEN A BARK AND A GROWL!  WHEN A
DOG BARKS HE THROWS HIS HEAD HIGH.  A BARK IS NOT A WAR CRY.
BUT WHEN A DOG GROWLS, HE ---3--- HIS HEAD.  A GROWL MAY MEAN
HE IS READY TO FIGHT. THE DOG MUST GUARD HIS THROAT WHEN HE
FIGHTS ANOTHER ANIMAL. IF A DOG COMES BARKING TO MEET YOU, YOU
ARE IN NO ---4---.



3A  HEADS
3B  RAISES
3C  LOWERS
3D  LOOSSES



***
```

The contents of locate, to use SANDERS notation is:

RES	MOV	V	V		V	FRM
CUR	CUR	T	T	DEL	T	TYP

Note This locate string works most of the time, but sometimes fails. I think it is a timing problem. There should be additional null characters between functions. Detailed timing is contained in the Appendix to the 720 manual.

Among the improvements in JCRT possible are:

Attaching it to the system, probably in early exploration during one of the first few sessions.

Storing the results of the reading test.

Obviously neither of these has practical significance unless the script network recognizes and differentiates among inquirers on the basis of this characteristic.

COST

COST represents a concept script overview. To me it is the way in which all our conceptual material should be presented. The overview flow chart represents the general organization of COST quite well.

The script contains all the material formerly in the scripts ROOMANDBOARD, HOWTOMEETCOLLEGECOSTS, COOPERATIVEWORKPROGRAMS, TUITIONATCOLLEGE, SCHOLARSHIPS, BOOKSSUPPLIESANDFEES and LIVINGEXPENSESATCOLLEGE. An example of the script is given in this protocol.

COLLEGE COST

INQUIRER: ISN'T THERE A SPECIAL GOVERNMENT LOAN FOR TEACHERS?

SYSTEM: THERE IS AN EDUCATIONAL LOAN THAT IS SPECIALLY TAILORED FOR PEOPLE WHO WANT TO TEACH. THAT IS THE NDEA LOAN. FOR EVERY YEAR YOU SERVE AS A FULL-TIME TEACHER IN A PUBLIC ELEMENTARY

SYSTEM: OR SECONDARY SCHOOL TEN PERCENT OF YOUR NDEA LOAN WILL BE CANCELLED. THIS IS GOOD UP TO A MAXIMUM OF 5 YEARS OR 50 PERCENT OF THE LOAN.

INQUIRER: HOW CAN I GET AN NDEA LOAN?

SYSTEM: YOU ARE ELIGIBLE FOR AN NDEA LOAN IF YOU ARE

A. ENROLLED FOR AT LEAST HALF THE NORMAL FULL TIME ACADEMIC WORKLOAD
B. CAPABLE, IN THE OPINION OF YOUR SCHOOL, OF MAINTAINING A GOOD ACADEMIC STANDING
C. IN NEED OF A LOAN TO CONTINUE YOUR COURSE OF STUDY AT THE INSTITUTION.

INQUIRER: DO YOU HAVE TO APPLY TO THE GOVERNMENT TO GET AN NDEA LOAN?

SYSTEM: IF YOU WANT TO APPLY FOR AN NDEA LOAN, APPLY TO THE SCHOOL YOU WISH TO ATTEND, NOT THE GOVERNMENT. CONSULT YOUR GUIDANCE COUNSELOR OR PRINCIPAL ABOUT THE LOAN AND CONTACT THE APPROPRIATE OFFICIAL OF THE COLLEGE.

INQUIRER: WHAT OTHER KINDS OF LOANS CAN I GET?

SYSTEM: DO YOU HAVE ANY (OTHER) QUESTIONS ABOUT NDEA LOANS?

INQUIRER: NO. WHAT ABOUT SOME OTHER LOANS?

SYSTEM: HOW ABOUT BANK LOANS, LOANS GIVEN BY THE STATE OF MASSACHUSETTS, GOVERNMENT LOANS, OR LOANS INTENDED FOR VOCATIONAL EDUCATION? DO ANY OF THESE INTEREST YOU?

INQUIRER: NO

SYSTEM: WHAT ARE YOU INTERESTED IN?

INQUIRER: HOW DO PEOPLE WHO CAN'T AFFORD IT, PAY FOR COLLEGE?

SYSTEM: YOU'VE ASKED HOW TO PAY FOR COLLEGE. THERE ARE LOTS OF DIFFERENT SOURCES OF FUNDS. HERE'S A LIST WHICH IS PROBABLY NOT EXHAUSTIVE, BUT QUITE COMPLETE.

1. SCHOLARSHIPS
2. LOANS

SYSTEM: 3. COLLEGE WORK-STUDY PROGRAMS
4. COLLEGE ASSISTANTSHIPS
5. PART TIME JOBS
6. TUITION PAYMENT PLANS
7. CO-OPERATIVE PLANS
8. VETERANS BENEFITS
9. FINANCIAL HELP FROM PARENTS

INQUIRER: WHAT ARE ASSISTANTSHIPS?

SYSTEM: COLLEGES GRANT ASSISTANTSHIPS TO STUDENTS (PARTICULARLY JUNIORS AND SENIORS) TO HELP PROFESSORS AND ADMINISTRATORS IN THE COLLEGE. THERE ARE USUALLY BASED ON NEED, ABILITY AND POTENTIAL. THE FINANCIAL AID OFFICE CAN PROBABLY TELL YOU ABOUT THE AVAILABILITY OF ASSISTANTSHIPS.

INQUIRER: YES, BUT WHAT DO ASSISTANTS DO?

SYSTEM: YOU'VE ALREADY SEEN ALL THE INFORMATION WE HAVE ON THAT SUBJECT. WHAT OTHER QUESTIONS DO YOU HAVE PERTAINING TO COLLEGE COST?

INQUIRER: TELL ME SOMETHING ABOUT CO-OP PROGRAMS.

SYSTEM: ARE YOU INTERESTED IN COOPERATIVE WORK PROGRAMS?

INQUIRER: YES.

SYSTEM: CO-OP PROGRAMS ARE A SPECIAL TYPE OF WORK STUDY PROGRAM. INSTEAD OF HOLDING PART TIME JOBS ALL YEAR LONG, STUDENTS SPEND SEMESTERS (OR TRIMESTERS) ALTERNATELY WORKING AND STUDYING. PART OF EACH YEAR IS SPENT WORKING FULL TIME, IDEALLY ON A JOB RELATED TO THE STUDENT'S MAJOR FIELD.

INQUIRER: HOW MUCH MONEY CAN THEY MAKE THAT WAY?

SYSTEM: HOW MUCH MONEY YOU MAKE ON A CO-OP JOB IS VARIABLE. IT CAN BE VERY LITTLE IF YOU HAVE A LOW SKILL JOB LIKE CLERKING. HOWEVER, IF YOU HAVE A SKILL, OR GAIN ONE DURING COLLEGE (LIKE PROGRAMMING OR KEYPUNCHING) THERE'S A POSSIBILITY OF MAKING QUITE A BIT OF MONEY.

The cost script includes a set of flags to indicate the state of inquiry and a set of RECOMPS, a bit string in which each bit means a particular piece of text has been seen. A list of these bits follows:

MEANING OF BITS IN FLAGS

Number	
1	Anyrecomp
2	
3	
4	
5	
6	Flag 1,Ø Mention of Geographical Area
7	Flag 1,Ø
8	Flag 1,Ø 1 = coming from RAB step 464.9Ø
9	Flag 1,Ø 1 = Yes, I was asking about data access (in NUND section)
1Ø	Flag 1,Ø 1 = Kid is coming from 14Ø.ØØ to look at questions again to pick one that he wants answers to. Ø = seeing questions for first time and going straight through program.
11	Flag 1,Ø From "NUND" area, we think kid is in wrong place. He isn't interested in cost. However, this flag will allow us to check one more time without subjecting him to any more rhetoric.
12	Kid has seen first set of questions. Wants to see more questions and get answers.
13	Kid has seen first set of questions, and may want second set. Ask him.
14	Kid has seen "unsure" question on NDEA loans and wants to branch to NDEA in loan section.
15	In scholar script, kid decided to see specific sources list.

MEANING OF BITS IN WORDS

Number	
1	Scholarship
2	Loans
3	Part time work, work study
4	Books, supplies, and fees
5	Parents pay
6	Room and Board
7	Tuition
8	How to pay

COST SCRIPT

MEANING OF BITS IN RECOMPS VECTOR

Number	Description
1	State residents have lower tuition than non-residents. This can make a big difference.
2	Differences in resident and non-resident tuition as great as $4000, varies from place to place.
3	Examples of resident and non-resident tuition
4	Quality and cost of schools are not necessarily related although many high prestige schools have high cost.
5	Quality and cost: ex. Berkeley (for residents, high quality, low cost)
6	Quality and cost - There are better indices of quality than cost.
7	General shpiel for unsure - Tuition = charge for instruction varies widely if public, private, resident, non-resident.
8	Recording of General shpiel for guy who's already had it.
9	Tuition range (for unsure presently)
10	More shpiel for unsure kid. Small liberal arts college costs less $ than big university or smashing reputation.
11	Lovejoy's tuition at private colleges
12	Suggest you ask questions about $AB or how to meet costs.
13	Lovejoy's on public schools
14	You can see examples of public tuitions or go to DATA.
15	You've already seen the list of resident, non-resident tuition, how about DATA or something else.
16	To get access to DATA, type @DATA.
17	Tuition at small liberal arts college is less than at a large university.
18	Tuition at a large university is more than at a small liberal arts college.
19	Tuition at a $(hc) top notch, Ivy League, etc. college is likely to be quite high.
20	Is there something else you want to know about cost and a $(hc:2) school?
21	You've asked about private tuition, would you like examples or data?
22	Example of private tuition
23	Geographical area is maybe a definite variable in living expenses.
24	Not sure we understand. Are you asking about variations in tuition costs?
25	Say it a different way, maybe we'll understand.

Number	Description
26	Help seq: We tried to understand about variable and cost, but didn't. Try again.
27	What a groove! He is interested in cost variations.
28	Type of school with low tuition is state. Is there a school for which you'd like to know tuition?
29	Can you think of a school who's tuition you'd like to know?
30	What's the name. Apology about list.
31	
32	
33	
34	School with low tuition is Berkeley.
35	Schools with high tuition are Harvard, Yale, University of Michigan, Smith.
36	If you need inspiration, use book.
37	
38	
39	He's just said tuition twice.
40	Is it correct that you're asking about tuition at college?
41	Oops. Guess not, sorry we mis-understood you.
42	We mis-understood you again. Here are some offerings.
43	You're not interested in any offerings. Well, what does interest you? (goes to Coll MINOR)
44	Tuition is the charge for instruction blah, blah.
45	Help seq: It probably doesn't do much good to know variations; you probably want specifics or preference script.
46	What interests you now?
47	General cost shpiel. (Cost = TUN = RAB) You can adjust RAB.
48	
49	
50	General RAB shpiel
51	Probably the next most useful thing is the data file.
52	You sound unsure. Try DATA - if you don't like it, stop.
53	Since you don't want to go to DATA, suggest something.
54	The DATA script will
55	Lovejoy on RAB
56	Not too useful to have generalities about RAB, how about DATA?
57	
58	
59	
60	Meal plans separate cost of room from cost of board.
61	Just for example, University of Massachusetts food plan saves $178.00.
62	Help seq: separating meals allows you to eat out.
63	General shpiel on snacks

Number	Description
64	Help seq: Your eating preferences are a factor in cost of food.
65	Food is included in RAB at most places.
66	Help seq: Rewarding food costs are included in RAB (65).
67	To find out more about food, ask students.
68	General food. You've mentioned a food word.
69	Can you be more specific?
70	Some schools allow only dormitory living - others allow fraternities, sororities, off campus, etc.
71	Fraternities and sororities
72	Apartments
73	Living with a family
74	Commuters vs. residents scripts are available.
75	What would you like to do?
76	Living in the dormitory only can be a drag, check it out.
77	Mund 1: we can't understand. Say that again.
78	Mund 1: Help frame - we weren't able to understand your words, rephrase.
79	Mund 2: Were you saying "Data access"?
80	Unsure, Help frame: Do you want data about a certain school?
81	Use DATA-COL to find out what variables are appropriate.
82	Question Mund 3 - Is tuition a concern? Interested in RAB? Food? Ways to meet costs?
83	You've probably been sent to the wrong place, say something and we'll take it to the top.
84	Good grief, Charlie Brown.
85	There are sections on TIN, RAB, school, loans, type word you want.
86	Are you unsure about costs data or how to meet costs?
87	If we understand correctly, neither of these (86) is it.
88	Oops, sorry. One of those areas was it. Which one was it again?
89	Specifics - How to get to school, how much, cost of etc.
90	First set of questions for UNS
91	Why don't you try a change of pace? Data base or Preference Script.
92	Second set of questions for UNS
93	Kid has said he doesn't want more questions, but he does want answers. "Which ones?"
94	What would you like to do? If answers, type in number of question.
95	What is it that you are unsure about. Would you rephrase what you said?

Number	Description
96	You may find a list of questions useful.
97	Books cost about $5Ø.ØØ per semester. There are ways to save.
98	2 kinds of extra expenses - money, clubs
99	
1ØØ	You are eligible for an NDEA loan if you are...
1Ø1	Amount you can borrow on NDEA
1Ø2	Repayment: General NDEA shpiel
1Ø3	Teaching allows 1Ø% rebate per year.
1Ø4	Ghetto - 15% and you can pay off whole loan by teaching.
1Ø5	Service, Vista, Peace Corps - no payment, no interest
1Ø6	How to apply for NDEA
1Ø7	There is an educational loan specially tailored for people who want to teach - NDEA.
1Ø8	Are you interested in how one goes about applying for an NDEA loan?
1Ø9	
11Ø	You mentioned something about NDEA loans. Do you want to know how to apply...
111	Do you have any other questions?
112	You've seen most material on NDEA. Do you want other stuff on loans?
113	Guaranteed loan is ... General shpiel
114	Eligible for help loan
115	Amount help loans
116	Repayment of help loan
117	How to apply for a help loan
118	Higher Education Act
119	Vocational loan
12Ø	Government loans - general shpiel
121	General business about bank loans
122	State specific
123	Loan uncertainty
124	
124	You seem to be asking how people dress.
126	Are you asking about dress at a specific college?
127	Do you want to know about (rules)? What to buy? When?
128	Standards of dress vary from place to place.
129	Regulation book is available.
13Ø	No data on dress at specific schools
131	Some people feel they have to go out and buy a wardrobe...
132	You can't know about emphasis on dress unless you talk to somebody
133	Special supplies for certain courses

Number	Description
134	Ways to pay list
135	Section on assistantships
136	Tuition payment plans
137	Veterans benefits
138	Parents
139	No more information on parents
14Ø	Employment - are you interested in going to college and working part time?
141	Working part time at college
142	Work-study
143	Work-study #2
144	Co-op programs #1
145	Antioch and Northeastern
146	How much do co-op programs cost?
147	Cost script contains the following things.

One improvement in the script would be to create certain storage for reference during later script usage. The RECOMPS vector should be stored combined with the earlier version. This would then be fetched and inquirer could begin a second interaction where he stopped the last time.

The COST script fails violently to deal with small talk answers. If an inquirer comments "interesting," the script uses specific and high order DECOMPS to decide if it is not a new question. It then goes directly to the not understood section. Flags and a counter could be used to recover to the previous minor section and to then probe for another question, perhaps even responding in some natural sounding way.

The COST script has been debugged for major loops, missing DECOMP lists and basic logic. It still fails frequently on fine points and on details of bookkeeping. Flags are never turned off, some counts are never reinitialized. This causes very strange behavior after extensive use of the script.

Because the script was so long, it was broken into two parts, COST and $COST. Communication between the two parts is through the execute command (EQ). No linkage is established. Normally a branch to an execute statement that transfers control to the other script is used rather than

a direct execute. This isolates and groups the XQ's from the embedding text making them easier to locate.

COST and several other scripts get into difficulty because of the @DATA command. When an inquirer uses @DATA and then types @STOP, he is returned to the CRT statement immediately preceding the KEYBOARD at which he typed @DATA. Sometimes the return point has text entirely inappropriate for his return. This is a system programming problem, not one for script authors, although it pays to be aware of what is going to happen to inquirers.

Occupational Preference Script

This script is our model of preference scripts. It embodies hundreds of hours of careful thought and thousands of clerical hours as well as several days of computer processing. It was intended to be a crude testing ground for several theoretical ideas; later modifications would smooth out the processing and incorporate the information derived from feedback in the field. Other preference scripts do not embody the same amount of thought and are generally weaker. The following suggestions apply to all preference scripts:

Specifications for Preference Scripts

Categories should be eliminated if the data are incomplete. If more than ten percent of the items have no information, the category should be eliminated.

The values within a category should each contain a significant fraction of the data items. No value with fewer than five percent should be included. Twenty percent to eighty percent is the best range. As an example, I would eliminate Jewish affiliated schools and non-allergenic workers from the preference scripts.

Generally, the broadest possible inclusion should be made. For example, in the occupational preference one choice is between business work and scientific work. The specification if a worker wants business work is to eliminate scientific jobs. This is correct since it includes all non-objectionable jobs. This makes a significant difference in the operation of scripts.

Note Occupational preference is the only one specified correctly by this criterion; however, it is coded incorrectly. Some examples of the size of level one records follow. Both the size according to current coding and the correct size according to script specification are given where they are available.

OCCUPATIONAL PREFERENCE
LEVEL 1 RECORDS

Script designation	Actual name	Number of items currently coded	Number in correct coding
A1	working with objects or things	479	733
A2	working with people and communicating ideas	131	385
B1	doing business work	148	731
B2	doing scientific and technical work	133	716
C1	routine, organized work	292	782
C2	unstructured, creative work	82	572
D1	working in social situations	38	346
D2	working with machines and processes	518	826
E1	getting praise and prestige from others	126	705
E2	working for your own satisfaction	159	738
G1	high in general intelligence	252	612
G2	low	152	512
G3	average	418	not available
V1	very good at words and language	231	766
V2	not good	298	633
V3	average	292	not available
N1	very good at math	138	not available
N2	not at all good	not chosen	726
N3	average	340	not available
P1	high in visual and artistic perception	205	458
P2	low	426	659
P3	average	500	not available
C1	high in ability to do clerical work	75	360
C2	low	524	789
C3	average	223	not available
M1	high in ability to work with hands and fingers	122	not available
M2	low	not chosen	742
M3	average	608	not available

Some of the lists derived from choices made in the occupational preference script are presented below. The categories above each list indicate the choices made. They can be interpreted according to the table given immediately above.

A1 B2 C2

14 of 24 jobs

organic chemist
biologist
geologist
forester
physicist
range manager
geophysicist
biochemist
soil scientist
soil conservationist
botanist
microbiologist
zoologist
agronomist

A1 B2

37 of 52 jobs

organic chemist
electrician
accountant
commercial airline pilot
draftsman
laboratory assistant
medical technologist
biologist
geologist
pharmacist
surveyor
forester
range manager
geophysicist
biochemist
health physicist
radiographer
loan officer
dairy farmer
meat poultry man
cattle rancher
cash grain farmer
vegetable grower
orchardist
soil scientist
physical oceanographer
metallurgist assistant
plant engineer
architectural draftsman
photogrammetrist
auditor
underwriter
navigator
chemical laboratory technician
medical laboratory assistant
assayer

A1 C1 D2 G3 V2 N3 M1

arc welder
gas welder

A2 B2

11 of 15 jobs

economist
occupational therapist
market-research analyst
anthropologist
historian
political scientist
sociologist
technical publications editor
nutritionist
scientific linguist
psychometrist

A2 B2 C2

economist
market research analyst
anthropologist
historian
political scientist
sociologist
scientific linguist
psychometrist
job analyst

A2 C2

13 of 30 jobs

actor
illustrator
industrial designer
literary writer
dancer
lawyer
judge
instrumental musician
economist
interior designer decorator
market-research analyst
popular singer
concert singer

A2 C2 E1

actor
literary writer
dancer
lawyer
judge
instrumental musician
popular singer
concert singer
continuity writer
critic
playwright
copywriter
art teacher

A2 C2 G1 P3

actor
literary writer
dancer
instrumental musician
interior decorator designer
political scientist
critic
job analyst
copywriter

A2 C2 G1 M1

illustrator
industrial designer
dancer
instrumental musician
clothes designer
stage scenery designer
painter
commercial designer
art teacher

A2 C2 G1 V1 M1

industrial designer
lawyer
judge
economist
market research analyst
psychometrist
job analyst

A2 E2

cook

A2 G1 M1

illustrator
industrial designer
dancer
secretary
stenographer
instrumental musician
clothes designer
stage scenery designer
athletic coach
painter
commercial designer
coroner
art teacher

A2 G1 + education highest

12 of 58 jobs

industrial designer
college faculty member
secondary school teacher
elementary school teacher
clergyman
branch manager
case worker
system engineer
lawyer
judge
librarian
economist

A2 N1 M1

cook
industrial designer

B1 C1 G3 V3 N3

12 of 18 jobs

store manager
sales clerk
cashier
post office clerk
service station attendant
mortgage clerk
airplane dispatch clerk
ticket agent
hotel clerk
station ticket agent
music librarian
bartender

B1 D1 G3

liscensed nurse
medical assistant

B1 E2

electrician
commercial airline pilot

B2 D1

chiropractor
liscensed nurse
registered nurse
optometrist
general practicioner
physical therapist
veterinarian
radiologic technologist
occupational therapist
speech therapist
speech pathologist
dental hygienist
osteopathic physician
county agricultural agent
home demonstration agent
nutritionist
audiologist
medical assistant
underwriter

B2 D2 G1 V3 N3

electrician
audio operator
medical laboratory assistant

B2 G1 N1

organic chemist
electrician
accountant
commercial airline pilot
architect
aeronautical engineer
chemical engineer
industrial engineer
draftsman
electronic technician
mechanical engineer

C1 D2

mail carrier
post office clerk
medical record librarian
mortgage work
airplane dispatch clerk
ticket agent
hotel clerk
music librarian
inventory clerk
file clerk
tile shader
order clerk

C2 D2

cosmetologist
barber
make-up artist
electrologist

E1 G1 P1 M1

dancer
chiropractor
dentist
general practicioner
veterinarian
osteopathic physician
film librarian
athletic coach
coroner
tugboat captain
art teacher

E2 G3 V3 N3 P2 C1 M3

cook

Inquirers should be able to look at their list at any time and return to the last choice point to continue their processing. This requires storing the step number of the choices.

Inquirers should be able to return to the previous value of their lists if they select too many choices at any point. This requires a temporary saving of the list before processing of each question. When all items are eliminated, one option should be to return to the last question and try again.

The logic employed at each choice, whether "Boolean or" or "Boolean and" should be clear from the question wording. Will a choice be included if it has any of the selections or must it have all of the selections?

It should be possible to allow sophisticated inquirers to choose the "or" or "and" option for themselves.

Some consistent use of send block should be established. I would use it for no preference and allow some other simple input to mean "keep the choices of last time."

In presenting the final list, it should be possible to show slides illustrating the options as well as verbalizing them. However, the #(SLIDE: :) command will be inappropriate since it requires separate run button pushes for each slide. It should be possible to use several of the component parts

of the command, changing only the part that turns off the projector \SØØ//////*.

An enormous convenience would report the choices made as well as the list derived. This would strengthen the association and help inquirers to remember what choices were made. Probably the easiest record keeping is a long vector with a position for each possible option in the script which would be decoded at the end.

It should be possible to control the flow of the preference script by a vector of factor choices created by ANALYZE. The questions presented would be derived from the vector rather than direct questioning. See ANALYZE script and tables 16-19 of the INQTABLE file for details.

It should be possible to let the inquirer order by importance to him the questions presented.

In storing away choices, the send block option should store as "no preference" instead of an incomplete option.

Inquirers often error by omitting spaces from between choices. Perhaps these responses could be processed or the script could more effectively teach him to insert spaces.

The wording of questions seems to make a difference. During all testing no one choice the option "not at all good at math" while several chose "not good in verbal ability." A random test, presenting different options at different times would help see if these options really differ and how wording effects response.

Order of options within a question seems to effect the likelihood of choice. This is another possible area for experimentation. A random function could assign order and records could be kept on logtape or on disc if that were deemed useful.

While an inquirer should gain some experience with the script by himself, it would be useful to provide special information after a few uses. As an example, an instance of real learning occurs as inquirer decreases the number of his choices to increase the size of his list. As he begins to use the "LIST" option, we see further learning. Once these two stages are

passed (they are easily recognized by a Monitor examination of inputs), we should provide him the record of all cumulative experience with the script. Several levels are possible:

1) We could include with each option the number of items included by it. This calculation may sometimes be complex because the directions do not always involve only one record.

2) We can provide a table that includes results of each possible pair of choices in a question. This is not always feasible, but would look like this:

	A2	B1	B2	C1	C2	D1	D2	E1	E2
A1	X		52	201	24				
A2	X		15		30	18			1
B1		X	X	48					2
B2			X		57	18	35		
C1				X	X		13		
C2					X		4		Ø
D1						X	X		
D2							X		

These are actual results of the 1969 field test for the first question of occupational preference. Such data provides a way of helping to avoid problems and speeding up the flow of information to the inquirer.

A coding observation: The actual coding of the script is rather inefficient since it does not make use of the history of other inquirers with the script. The same processing and fetches are done for all inquirers. At least with the early questions, when few combinations are possible, we could create new level one records during the interaction and fetch them rather than the raw records. As one example, the pair of choices A2, B2, C1 came up repeatedly. A single fetch could provide the needed data with slightly greater cost of internal processing.

I would suggest that the DECOMPS could lead to a bit vector, each bit corresponding to a choice. For example, mapping

A1 → 1 A2 → 2 B1 → 3 B2 → 4

C1 → 5 C2 → 6 D1 → 7 D2 → 8

E1 → 9 E2 → 1Ø

the choice alone would be coded Ø1Ø11ØØØØØ. This coding provides a unique vector for each set of choices. The vector could be used as the name for the combined bit vector associated. Logic would be as follows:

1) encode the choices as a bit vector.
2) fetch for that record name: Binary 1.
   2a) if found, you are done.
   2b) if not found, process as usual, one choice at a time and store the result under the bit vector name.

It is possible to use subparts of the bit vector (say the first two on bits only) to provide fewer vectors at higher processing cost. If this is used, the data base must have space left for this expansion. In recording this, be sure to consider the possibility of packing more than one level one record into a physical record on the disc by assigning different descriptor names. This could have major repercussions on other scripts (DATA-HELPNAME), but could save access time.

Aylmer has observed that some inquirers differentiate between personally and objectively worded questions. The occupational preference script concentrates on personal evaluations: I am not good in math. The college preference script is objective: the school has a radio station. Presumably each question could be worded either way; obviously these wordings do not have the same emotional loading. Experiments should be able to distinguish the effect of each wording and offer some useful insight into the inquirers psychological mechanisms.

One extension of the preference script idea would be to write a script that immediately gave feedback on the consequences of choices and facilitated comparisons of lists with each other. What are the consequences of each choice? How do the choices interact? What are the items on one list eliminated by a later choice? All of these questions could be answered for an inquirer by a rapid acting routine. This experience

would familiarize him with the preference script idea and enable him to deal more effectively with the many dimensions available to him.

Such a script would present choices with a minimum of verbal material. The choices would be encoded and the final list associated with choices. After several lists were available, the machine, operating as Monitor, would compare the choices made and the lists received. Both their content and the differences would be demonstrated to the inquirer.

There are two possible approaches: one is to use the encoded vectors as logical symbols; the other is to build meaning into the script. As logical symbols you would expect the inquirer to supply the meaning for the constructs. Such a routine could treat any set of questions and lists and is therefore economical of coding effort.

The second approach would employ the external world understanding of the concepts to make comments on the specific choices made by inquirers. It might comment on inconsistent choices and notice the dramatic reduction in list size that results.

There is a way to integrate the two approaches. The first described script could compute the differences and similarities of choice vectors and result vectors, leaving the presentation to another script.